

**No title available**

?????

Publication number: ()

Publication date:

Inventor(s):

Applicant(s):

Classification:

- international:

- European:

Application number:

Priority number(s):

Abstract not available for ()

Abstract of corresponding document: **WO 9957839 (A2)**

Method and system for performing a service on a home network having a plurality of home devices connected thereto, by: connecting a client device to the home network for displaying a user interface; executing a software agent on the client device for obtaining selection information for the network devices and displaying the selection information on a user interface displayed on the client device; selecting a first home device connected to the network from the user interface being displayed on the client device; reading first capabilities data for the first home device, where the first capabilities data includes information in a structured format for identifying the capabilities of the first home device; reading second capabilities data for a second home device connected to the network, where the second capabilities data includes information in the structured format for identifying the capabilities of the second home device; comparing the first and second capabilities data of the first and second home devices, respectively; selecting the second home device from the user interface displayed on the client device; and sending control and command data from the client device to the first and second home devices to cause the first and second home devices to communicate with each other to perform the service.

Data supplied from the **esp@cenet** database — Worldwide

(19) 日本国特許庁 (J P)

(12) 公表特許公報 (A)

(11) 特許出願公表番号  
特表2002-514797  
(P2002-514797A)

(43) 公表日 平成14年5月21日 (2002.5.21)

(51) Int.Cl. <sup>7</sup>	識別記号	F I	テームト* (参考)
G 0 6 F 13/00	3 5 7	G 0 6 F 13/00	3 5 7 A 5 B 0 8 9

審査請求 未請求 予備審査請求 有 (全 95 頁)

(21) 出願番号 特願2000-547722(P2000-547722)  
 (86) (22) 出願日 平成11年5月7日(1999.5.7)  
 (85) 翻訳文提出日 平成12年11月7日(2000.11.7)  
 (86) 国際出願番号 P C T / K R 9 9 / 0 0 2 2 1  
 (87) 国際公開番号 W O 9 9 / 5 7 8 3 7  
 (87) 国際公開日 平成11年11月11日(1999.11.11)  
 (31) 優先権主張番号 6 0 / 0 8 4 , 5 7 8 Equivalent to Ref. 1  
 (32) 優先日 平成10年5月7日(1998.5.7)  
 (33) 優先権主張国 米国 (U S)

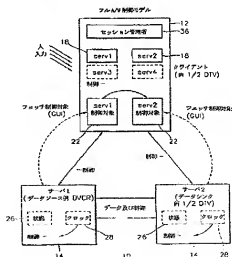
(71) 出願人 サムスン エレクトロニクス カンパニー  
リミテッド  
大韓民国 キュンキード スオン市 バル  
ダルク マエタンードン 416  
 (72) 発明者 リチャード・ハンブルマン  
アメリカ合衆国・カリフォルニア・  
94539・フリモント・ロウアー・ヴィン  
ターズ・サークル・343  
 (72) 発明者 ドンギャン・ワン  
アメリカ合衆国・カリフォルニア・  
95054・サンタ・クララ・オーク・グロー  
ブ・ドライブ・#302・440  
 (74) 代理人 弁理士 志賀 正武 (外 1 名)

最終頁に続く

(54) 【発明の名称】 ネットワークで汎用的にアクセスする命令及び制御情報のための方法及び装置

## (57) 【要約】

第1ホームデバイスは第1アプリケーションインタフェースデータを貯蔵し、第2ホームデバイスは第2アプリケーションインタフェースデータを貯蔵する。データベースは、データベースデバイスに第1及び第2ホームデバイスのための前記アプリケーションインタフェース説明データを伝送するために第1及び第2ホームデバイスを問い合わせることによって形成される。前記データベースは、データベースデバイスに貯蔵されたり、ネットワークデバイスにより汎用的なアクセスのためのネットワークに接続されたりする。従って、前記第1ホームデバイスのための第1アプリケーションインタフェース説明対象はネットワークを通じてデータベースから第2ホームデバイスに提供される。さらに、前記第2アプリケーションインタフェース説明対象はネットワークを通じてデータベースから第1ホームデバイスに提供される。



**【特許請求の範囲】**

**【請求項1】** ホームネットワーク上でサービスを行うための方法において

- (a) 第1ホームデバイスを前記ホームネットワークに接続する段階と、
- (b) 第2ホームデバイスを前記ホームネットワークに接続する段階と、
- (c) 複数のアプリケーションインタフェース説明データ対象を含むデータベースを提供する段階として、各アプリケーションインタフェース説明データ対象は前記ネットワークに接続された一つまたはそれ以上のホームデバイスにより構造化されたフォーマットでホームデバイスの命令及び制御のための情報を含む段階と、
- (d) 前記第2ホームデバイスが前記データベースで前記第1ホームデバイスのための第1アプリケーションインタフェース説明対象をアクセスする段階と、
- (e) 前記第1ホームデバイスが前記データベースで前記第2ホームデバイスのための第2アプリケーションインタフェース説明対象をアクセスする段階と、
- (f) 前記ネットワークを通じて前記第2デバイスのために前記アプリケーションインタフェース説明対象を使用して前記第1ホームデバイスから前記第2ホームデバイスに命令及び制御データを伝送する段階と、
- (g) 前記ネットワークを通じて前記第1デバイスのために前記アプリケーションインタフェース説明対象を使用して前記第2ホームデバイスから前記第1ホームデバイスに命令及び制御データを伝送する段階を含み、それにより前記第1及び第2ホームデバイスが前記サービスを行うことを特徴とするホームネットワーク上でサービスを提供する方法。

**【請求項2】** 前記構造化されたフォーマットはXMLフォーマットを含むことを特徴とする請求項1に記載の方法。

**【請求項3】** 前記(c)段階はデータベースデバイスを前記ネットワークに接続する段階であって、前記データベースデバイスは前記データベースを含むことを特徴とする請求項1に記載の方法。

**【請求項4】** (i) 前記第1ホームデバイスは前記第1アプリケーションインタフェースデータを貯蔵し、

(ii) 前記第2ホームデバイスは前記第2アプリケーションインタフェースデータを貯蔵し、

(iii) 前記(c)段階は、前記データベースデバイスに前記第1及び第2ホームデバイスのための前記アプリケーションインタフェースデータを伝送するために前記第1及び第2ホームデバイスを問い合わせることを含む段階により前記データベースを形成する初期化段階を含むことを特徴とする請求項3に記載の方法。

【請求項5】 前記(d)段階は、ネットワークを通じて前記データベースから前記第2ホームデバイスに前記第1ホームデバイスのための第1アプリケーションインタフェース説明対象を提供する段階を含むことを特徴とする請求項1に記載の方法。

【請求項6】 前記(e)段階は、ネットワークを通じて前記データベースから前記第1ホームデバイスに前記第2ホームデバイスのための第2アプリケーションインタフェース説明対象を提供する段階を含むことを特徴とする請求項1に記載の方法。

【請求項7】 三つまたはそれ以上のホームデバイスを前記ネットワークに接続する段階として、少なくとも一つのホームデバイスは前記ネットワークを通じて複数のホームデバイスに命令及び制御データを伝送するための複数のホームデバイスのアプリケーションインタフェース説明対象を問い合わせるためにデータベースをアクセスする段階をさらに含むことを特徴とする請求項1に記載の方法。

【請求項8】 各アプリケーションインタフェース説明対象は構造化されたフォーマットでデータを含むことを特徴とする請求項1に記載の方法。

【請求項9】 サービスを提供するためのネットワークシステムにおいて、

(a) 相互通信するデバイスにより使われる通信媒体を提供する物理層と、

(b) 第1ホームデバイスと、

(c) 第2ホームデバイスと、

(d) それぞれのアプリケーションインタフェース説明対象は、前記ネットワークに接続された一つまたはそれ以上の他のデバイスにより構造化されたフォーマットでホームデバイスの命令及び制御のための情報を含む多数のアプリケーション

インタフェース説明データ対象を含むデータベースとを含み、

前記第2ホームデバイスは、前記データベースで前記第1ホームデバイスのための第1アプリケーションインタフェース説明対象をアクセスし前記第1アプリケーションインタフェース説明対象を使用して、前記第2ホームデバイスから前記第1ホームデバイスに制御及び命令データを伝送するためのアプリケーション制御手段を含み、

前記第1ホームデバイスは、前記データベースで前記第2ホームデバイスのための第2アプリケーションインタフェース説明対象をアクセスし前記第2アプリケーションインタフェース説明対象を使用して、前記第1ホームデバイスから前記第2ホームデバイスに制御及び命令データを伝送するためのアプリケーション制御手段を含み、

それにより前記第1及び第2ホームデバイスは前記サービスを行うことを特徴とするサービスを提供するためのネットワークシステム。

【請求項10】 前記構造化されたフォーマットはXMLフォーマットを含むことを特徴とする請求項9に記載のネットワークシステム。

【請求項11】 前記データベースを貯蔵するデータベースデバイスをさらに含むことを特徴とする請求項9に記載のネットワークシステム。

【請求項12】 (i) 前記第1ホームデバイスは第1アプリケーションインタフェース説明対象を貯蔵し、

(ii) 前記第2ホームデバイスは第2アプリケーションインタフェース説明データを貯蔵し、

(iii) 前記データベースデバイスは、データベースデバイスに各々前記第1及び第2アプリケーション説明対象を伝送するために前記第1及び第2ホームデバイスを問い合わせることによって前記データベースを形成することを特徴とする請求項11に記載のネットワークシステム。

【請求項13】 前記第2ホームデバイスの制御アプリケーション手段は、前記データベースから第1ホームデバイスのための第1アプリケーションインタフェース説明対象を得ることを特徴とする請求項9に記載のネットワークシステム。

【請求項14】 前記第1ホームデバイスの制御アプリケーション手段は、前記データベースから第2ホームデバイスのための第2アプリケーションインタフェース説明対象を得ることを特徴とする請求項9に記載のネットワークシステム。

【請求項15】 少なくとも一つのホームデバイスは、ネットワークを通じて複数のホームデバイスに命令及び制御データを伝送するための複数のホームデバイスのアプリケーションインタフェース説明対象を問い合わせるためにデータベースをアクセスする三つまたはそれ以上のホームデバイスをさらに含むことを特徴とする請求項9に記載のネットワークシステム。

【請求項16】 それぞれのアプリケーションインタフェース説明対象は構造化されたフォーマットでデータを含むことを特徴とする請求項9に記載のネットワークシステム。

【請求項17】 前記構造化されたフォーマットはXMLフォーマットを含むことを特徴とする請求項9に記載のネットワークシステム。

## 【発明の詳細な説明】

## 【0001】

## 【発明の属する技術分野】

本発明はネットワークシステム分野に係り、より詳細にはそれに連結された複数のデバイスを具備したホームネットワークに関する。

## 【0002】

## 【従来の技術】

一般的に、ネットワークは通信リンクと前記通信リンクと接続された通信能力を有する多様なデバイスを含む。前記デバイスは、コンピュータ、周辺装置、ルータ、貯蔵装置、そしてプロセッサと通信インタフェースを具備した製品を含む。ネットワークの一実施形態として多様なデバイスが相互接続された家庭用ホームネットワークを挙げられる。普通の家庭はパソコン及び家庭内で典型的に見つけられるホームデバイスを含むいろいろなデバイスを含むことができる。用語としての“デバイス”は、典型的に論理デバイスまたは機能性及びデータを交換する能力を備える他の装置を含み、全てのホームデバイスだけでなく汎用コンピュータも含むことができる。ホームデバイスは保安システム、映画館装備、TV、VCR、ステレオ装備、デジタル衛星サービス(DSS: Digital Satellite Services)とも知られた直接放送衛星サービス(DBSS: Direct Broadcast Satellite Services)、散水システム、照明システム、電子レンジ、食器洗い器、オープン/ストーブ、洗濯機/乾燥器及び自動車内の処理システムのような電子デバイスを含む。

## 【0003】

一般にホームデバイスは家主のライフスタイルと生活水準を向上させるために使われる。例えば、食器洗い器で汚い食器を洗浄することによって、家主が手で直接食器を洗わなくて済む。VCRでは後で家主が特定のプログラムを見るようにTVプログラムを録画できる。保安システムは家主の貴重品を保護し、望ましくない侵入者による家主の不安が軽減される。

## 【0004】

家庭用映画館装備のようなホームデバイスは、よく単一の共通制御ユニット、即ち遠隔制御装置を使用して制御される。前記単一の共通制御ユニットは家主が

単一のインタフェースを使用している他のホームデバイスを制御し命令できるようにする。従って、製造者は単一インタフェースからそのホームデバイスを制御し命令するための制御ユニットを発展させてきた。

#### 【0005】

ホームデバイスを制御し命令する遠隔制御ユニットの使用に関する短所は各ホームデバイスを制御し命令するための静的な命令ロジックを提供するということである。遠隔制御ユニットの使用に関した他の短所は、既知の遠隔制御ユニットが複数の多様なデバイスを制御できないということであって、より詳細には作業を行ったりまたはサービスを提供するために相互通信するための相異なる互換性を有する複数のデバイスを制御できないということである。

#### 【0006】

従来のネットワークシステムで、使用者は遠隔制御ユニットまたはデバイス制御パネルを使用して命令を提供する。一応使用者が止めれば、ネットワークで自動操作のための命令を提供する制御ユニットとデバイスがなくなる。使用者が初期にデバイスの第1セットを制御し命令した後、従来のシステムは使用者がデバイスの第2セットを直接制御及び命令せずに作業を行うために必須のことである、ネットワークでデバイスの第2セットと自動で通信するデバイスの第1セットのためのメカニズムを提供しない。さらに、従来のシステムは多様なネットワークデバイスがネットワークで命令及び制御を行うために他のネットワークデバイスに対して情報を得る効率的な方法を提供しない。

#### 【0007】

##### 【発明が解決しようとする課題】

従って、ホームネットワークでデバイスの動的制御及び命令を提供する方法とシステムが必要である。また作業を行ったりまたはサービスを提供するために相互通信することにおいて相異なる性能を有する複数のデバイスを制御できる能力を提供するための方法及びシステムが必要である。また多様なネットワークデバイスがさらに他の多様なネットワークデバイスを自動で命令し制御する能力を提供するための方法及びシステムが必要である。またデバイス間通信のために汎用的にアクセスする命令及び制御情報を提供するための方法及びシステムが必要で

ある。

# 【0008】

## 【課題を解決するための手段】

本発明はこの必要性を満たす。第1実施形態で本発明はホームネットワーク上でサービスを行うための方法及びシステムにおいて、第1及び第2ホームデバイスを前記ホームネットワークに接続する段階と、複数のアプリケーションインタフェース説明データ対象を含むデータベースを提供する段階として、各アプリケーションインタフェース説明データ対象は前記ネットワークに接続された一つまたはそれ以上のホームデバイスにより構造化されたフォーマットでホームデバイスの命令及び制御のための情報を含む段階と、第2ホームデバイスが前記データベースで前記第1ホームデバイスのための第1アプリケーションインタフェース説明対象をアクセスする段階と、前記第1ホームデバイスが前記データベースで前記第2ホームデバイスのための第2アプリケーションインタフェース説明対象をアクセスする段階と、前記ネットワークを通じて前記第2デバイスのために前記アプリケーションインタフェース説明対象を使用して前記第1ホームデバイスから前記第2ホームデバイスに命令及び制御データを伝送する段階と、前記ネットワークを通じて前記第1デバイスのために前記アプリケーションインタフェース説明対象を使用して前記第2ホームデバイスから前記第1ホームデバイスに命令及び制御データを伝送する段階とを含む。それにより前記第1及び第2ホームデバイスは前記サービスを行う。

# 【0009】

本発明の一形態で、前記第1ホームデバイスは第1アプリケーションインタフェースデータを貯蔵し、前記第2ホームデバイスは第2アプリケーションインタフェースデータを貯蔵する。前記データベースは、前記データベースデバイスに第1及び第2ホームデバイスのための前記アプリケーションインタフェース説明データを伝送するために第1及び第2ホームデバイスを問い合わせることによって形成される。前記データベースは、データベースデバイスに貯蔵されたり、ネットワークデバイスにより汎用的なアクセスのためのネットワークに接続されたりする。従って、前記第1ホームデバイスのための第1アプリケーションインタ

フェース説明対象はネットワークを通じてデータベースから第2ホームデバイスに提供される。さらに、前記第2アプリケーションインタフェース説明対象はネットワークを通じてデータベースから第1ホームデバイスに提供される。

#### 【0010】

さらに、三つまたはそれ以上のホームデバイスはネットワークに接続され、少なくとも一つのホームデバイスはネットワークを通じて複数のホームデバイスに命令及び制御データを伝送するための複数のホームデバイスのアプリケーションインタフェース説明対象を問い合わせるためにデータベースをアクセスする。各アプリケーションインタフェース説明対象は構造化されたフォーマットでデータを有する。前記構造化されたフォーマットはXMLフォーマットを有する。

#### 【0011】

このような本発明の特徴、観点及び利点は次の詳細な説明、請求項及び添付した図面によりさらによく理解されるはずである。

#### 【0012】

##### 【発明の実施の形態】

一観点で、本発明はホームネットワークのようなネットワークでデバイス間通信を提供する。ホームデバイスがより知能的になって情報を共有できるようになれば、デバイス間通信は使用者が前記デバイスの情報共有性能を用いられるようにネットワーク内のデバイスを相互接続することを許容する。同じように、デバイス間通信は使用者にネットワークに連結されたデバイスを充分柔軟に使用する能力を与えることにおいて決定的な役割をする。

#### 【0013】

図1を参照すれば、本発明の一実施形態で、ネットワーク10は通信リンク16を通じて相互連結された少なくとも一つのクライアントデバイス12及び少なくとも一つのサーバデバイス14を含む。通信リンク16は、多様に連結されたホームデバイス間でデータを伝送し受信するために物理的階層(媒体)を提供する1394シリアルバスを含むことができる。前記1394シリアルバスは、時間-多重化オーディオ/ビデオ(A/V)ストリーム及び標準IP(Internet Protocol)通信を両方とも支援する。実施形態で、ホームネットワークはホームネットワーク

のための通信階層としてIPネットワーク階層を使用する。しかし他の通信プロトコルがホームネットワークに通信を提供するために使われる場合もある。

#### 【0014】

各クライアントデバイス12は、ネットワーク10内の一つまたはそれ以上のサーバデバイス14と通信できる。さらに、各サーバデバイス14は、ネットワーク10で一つまたはそれ以上の他のサーバデバイス14、そして一つまたはそれ以上のクライアントデバイス12と通信できる。各クライアントデバイス12は、使用者入力を受信するためのマウスとキーボードのような入力装置を使用者がネットワークに連結されたデバイスと相互作用するように制御使用者インタフェースを提供するディスプレイを含む使用者通信インタフェースを含むことができる。使用者インタフェースは、使用者に情報を提供するためのGUI(Graphical User Interface)ディスプレイ18を含むことができる。図2を参照すれば、ここに定義されたように、各サーバデバイス14は使用者に制御使用者インタフェースを除外したサービスを提供し、各クライアントデバイス12はネットワーク10と相互作用する使用者のために制御使用者インタフェースを提供する。このようにしてクライアントデバイス12だけが使用者と直接相互作用し、サーバデバイス14はクライアントデバイス12及びサーバデバイス14とのみ相互作用する。例えば、サービスはMPEGソーシング/シンキング及びディスプレイサービスを含むことができる。

#### 【0015】

図3は、多数のクライアントデバイス12及び多数のサーバデバイス14を含むホームネットワーク10の例を示すブロック図である。各サーバデバイス14は、ネットワークで使用者にサービスを提供するための資源としてハードウェアを含むことができる。さらに各サーバデバイス14はサーバハードウェアを制御するためのサーバまたはサービス制御プログラム20を貯蔵することができ、図4に示したようにサーバ制御プログラム20と共に使用者インタフェースのためのグラフィック制御対象(GCO: Graphical Control Object)使用者インタフェース説明書22を含むことができる。

#### 【0016】

制御クライアントデバイス12と制御されたサーバデバイス14との間の制御のために、クライアントデバイス12は、例えばネットワークを通じてサーバデバイス14からクライアントデバイス12にGC022を伝送することによってサーバデバイス14のGC022をアクセスする。その後、クライアントデバイス12は、ネットワークを通じてクライアントデバイス12からサーバデバイス14の制御プログラム20と通信を行う使用者のための制御使用者インタフェースGUI18を創出するために伝送されたGC022を使用する。使用者は少なくともクライアントデバイス12からサーバデバイス14の制御プログラム20に制御と命令を提供する。

#### 【0017】

サーバデバイス自体で各サーバデバイス14のGC022貯蔵は、いくつかのサーバデバイス14と共にネットワークでクライアントデバイス12のプロセッシングと貯蔵要求を縮められる。さらに、サーバデバイス14でGC022を貯蔵することは、各サーバデバイス14が使用者にその自体のGUI形態と感じを提供することを許容し、クライアントデバイス12で修正なしにGC022の修正または更新を許容する。

#### 【0018】

実施形態でクライアントデバイス12とサーバデバイス14との間の命令及び制御を提供する図4を参照すれば、前記クライアントデバイス12は、クライアントデバイス12に貯蔵されたり希望のサーバデバイス14からネットワークを通じてクライアントデバイス12に伝送されたGC022を使用してGUI18をディスプレイするためのレンダラ(Renderer)24を含むことができる。例えば、初期のデバイス選択状態で、前記クライアントサーバ12はネットワークを通じて少なくとも一つのサーバデバイス14のGC022をフェッチでき、前記レンダラ24は前記サーバデバイス14を制御するためにGC022を使用してGUI18をディスプレイする。望ましくは、前記GUI18はサーバデバイス14でカスタマイズされ、サーバデバイス14を制御するために作られた内蔵命令セットを含むことができる。

#### 【0019】

また、多様なサーバデバイス14のGUI18は、例えば：(1) GUI18をディスプレイするクライアントデバイスレンダラ24のための共通GCOモデルタイプ、(2) 多様なサーバデバイス14からクライアントデバイス12にGCO22を伝送するための共通通信プロトコル、(3) 制御される特定サーバデバイス14の対応するサーバデバイス14の制御プログラムで作られた知識を求めないクライアントデバイス12から対応するサーバデバイス14の制御プログラム20にGUI相互作用のための共通通信プロトコルのような共同体を含むことができる。

#### 【0020】

再び図4を参照すれば、サーバデバイス14は、サービスを提供するためにサーバハードウェアを制御する一つまたはそれ以上のサーバ制御プログラム20を含むことができる。前記サーバデバイス14のGCO22からGUIインタフェース18はサーバデバイス制御プログラム20にインタフェースを提供する。前記サーバデバイス14は、求められたサービスを提供することにおいてサーバデバイス14及びサーバデバイスハードウェアの制御状態を指摘する制御状態データ26を含むことができる。

#### 【0021】

例えば、制御状態データ26は、VCRサーバデバイスで録画動作を行うために設定されるタイマー設定のように、サーバデバイス14のためにGUI18で制御情報の状態を含むことができる。制御状態データ26は制御されたサーバデバイス14に貯蔵され、サーバデバイス14の使用者制御のために制御クライアントデバイス12でサーバデバイス14のGUI18を通じて使用者にディスプレイされる。望ましくは、前記サーバデバイス14のGUI18をディスプレイするための制御クライアントデバイス12は、制御されたサーバデバイス14のための制御状態データ26の知識を維持しない。

#### 【0022】

各サーバデバイス14は一つまたはそれ以上のクライアントデバイス12により制御される。従って、前記サーバデバイス14に貯蔵された制御状態データ26は、各制御クライアントデバイス12でサーバデバイス14のGUI18内の情報状態を含む。例えば、使用者が第1クライアントデバイス12を使用してサー

バデバイス14を制御する時、使用者制御が完成されれば、第1クライアントデバイス12でサーバデバイス14のGUI18内の情報はサーバデバイス14によりサーバデバイス14の制御状態データ26に貯蔵される。

### 【0023】

一方では、使用者が第1クライアントデバイス12でサーバデバイス14のGUI18と相互作用する間、前記サーバデバイス14の制御状態データ26は第1クライアントデバイス12でサーバデバイス14のGUI18内の情報に更新され、使用者制御が完成されれば、前記制御状態データ26はサーバデバイス14で維持される。使用者が第2クライアントデバイス12を使用してサーバデバイス14を制御する時、前記制御状態データ26は後続する制御のために第2クライアントデバイス12でサーバデバイス14のGUI18を通じて使用者が用いることができるように作られる。使用者はまた以後にサーバデバイス14を制御するために第1クライアントデバイス12を使用でき、ここで前記制御状態データ26は後続する制御のために第1クライアントデバイス12でサーバデバイス14のGUI18を通じて使用者が用いることができるように作られる。前記サーバデバイス14はまたクロック28を含んだり、後述されるように使用者から入力された時間またはクロックに基づいた時間遅延作用を許容するために現在時間を維持できる。

### 【0024】

クライアントデバイス12とサーバデバイス14は、例えばDTVのように一つのユニットで物理的に一緒に結びつけることができる。その場合に、前記サーバデバイス14はサーバハードウェアを制御するための制御プログラム20を含み、前記クライアントデバイス12は少なくともサーバハードウェアの制御及び命令のためにサーバ制御プログラム20に制御使用者インタフェースを提供する。図5はクライアントデバイス12の例を示す図面であって、(1) GUIディスプレイのためのPDA(RemoteC)、(2) GUIをディスプレイし、オーディオ及び/またはビデオプログラムストリーム目的地サーバで構成されたシンクサーバを含むDTV(STB)、(3) GUIをディスプレイし、多重のサービスを提供するために少なくとも一つのサーバデバイスを含むPCを含むことができる。DTVまたはPCクライアント

デバイス内のハードウェアあるいは実行できるものは他のクライアントデバイスによっても制御される。図6はサーバデバイス14の例を示す図面であって、(1) ソースサーバデバイスとしてのDVDP SmartCard、(2) シンクサーバデバイスとしてのオーディオ増幅器、(3) ソースまたはシンクサーバデバイスとしてのDVCR、(4) 遠隔サーバデバイスを管理するための管理サーバを含む。前記管理サーバは例えばDSB-STB、ケーブルTV-STB、またはATSC-STBを含むことができる。このようなデバイスはSTB内部作業のローカル制御または管理のための管理サーバを含む。さらに、外部ネットワークを通じてアクセスされた外部サーバは、例えばVideo-on-Demand、Enhanced-TV及びインターネット商取引のようなサービスのためにローカルクライアントデバイスにより使われる場合がある。

#### 【0025】

図7を参照すれば、二つのサーバデバイス14間の通信及び制御は、それら間の命令語及び制御データを通信するサーバデバイス14の制御プログラム20により行われる。サーバデバイス14はネットワークを通じて一つまたはそれ以上の他のサーバデバイス14を制御できる。そして、サーバデバイス14は一つまたはそれ以上のサーバデバイス14、そして一つまたはそれ以上のクライアントデバイス12により制御される。さらに、使用者はサーバデバイス14の第1セットを制御及び命令するためにクライアントデバイス12を使用でき、前記サーバデバイス14の第1セットは、使用者にサービスを行うために必須のものである。使用者を惑わせずにサーバデバイス14の第2セットを自動で命令し制御できる。

#### 【0026】

例えば、自動時間一遅延動作を行うために、使用者はサーバデバイス14の第1セットを制御するクライアントデバイス12に“ログオン”し、希望のサービスを指定できる。それから使用者は前記クライアントデバイス12から“ログオフ”する。サーバデバイス14の第1セットはそれら自体間で通信及び制御を行い、その後第1セットで一つまたはそれ以上のサーバデバイスが使用者の介入なしに希望のサービスを総括的に提供するために必須のものであって、サーバデバイス14の第2セットを自動で制御する。

## 【0027】

図7は、二つのサーバデバイス14が相互通信し制御する実施形態を示す図面である。各サーバデバイス14は前述された制御プログラム20、クロック28及び前述した制御状態データ26を含む。各サーバデバイス14はまたサーバデバイス14がクライアントデバイス12により直接制御されるようにするGC022を含むことができる。しかし、GC022はクライアントデバイス12により直接制御されなく、ただ他のサーバデバイス14と通信するサーバデバイス14には含まれる必要がない。各サーバデバイス14はまた命令語(CL: Command Language)インタフェース30及び命令語ライブラリを含む。前記命令語ライブラリは、前記サーバデバイス14がそのサービスを提供するために情報を送信及び受信するのに使用する命令を含む。しかし、図4及び前述されたように命令語は使用者制御のために必ず必要なものではない。

## 【0028】

図8は、ネットワークでソースサーバデバイス14、シンクサーバデバイス14、そしてクライアントデバイス12を含むオーディオ/ビデオ(A/V)モデルの例を示す。前記ソースサーバデバイス14はソースサーバデバイス14のデータストリームソースハードウェア32を制御するための制御プログラム20を含み、前記シンクサーバデバイス14はシンクサーバデバイス14のデータストリームシンクハードウェア34を制御するための制御プログラム20を含む。例示的な動作で、使用者はデータストリームソースハードウェア32を稼働させるためにソースサーバデバイス14を制御し、データストリームシンクハードウェア34を稼働させるためにシンクサーバデバイス14を制御するクライアントデバイス12を使用する。データストリームソースハードウェア32からデータストリームシンクハードウェア34へのデータ伝送を初期化することにおいて、使用者は前記クライアントデバイス12を諦める。他の一方では、使用者は将来の初期データ伝送をプログラムでき、クライアントデバイス12を諦める。以後でソースサーバデバイス14のデータストリームソースハードウェア32とシンクサーバデバイス14のデータストリームシンクハードウェア34は使用者によりプログラムされた時間に自動でデータ伝送を初期化する。

## 【0029】

例えば、前記データストリームソースハードウェア32は、直接衛星放送(DBS: Direct Broadcast Satellite)のようなチューナーアクセスデバイスを含むことができる。DBSはケーブルテレビに代える多重チャンネルであって、衛星から直接的に小さな衛星円盤(直径3フィート18インチ)にケーブルのようなテレビプログラミングを提供する。DBSと共に、いくつかの標準アナログテレビ信号は単一衛星トランスポンダでデジタル的に圧縮され、空中の指定された場所に位置した衛星円盤も200つ以上のチャンネルが受信することを許容する。前記データストリームシンクハードウェア34は、再生時に圧縮されたデジタルビデオ信号をデコーディングできる命令及びデジタルビデオカセットレコーダ(DVCR)を含むことができる。使用者はDVCRのための“時間-遅延録画”イベントデータとチューナーアクセスデバイスのための“時間-遅延選択プログラム”イベントデータを含む命令及び制御データを提供する。時間遅延後に前記チューナーアクセスデバイスは希望のプログラムを選択し、使用者から追加的な制御操作なしでもプログラムデータを受信及び録画するDVCRにソースプログラムデータを提供する。

## 【0030】

図9は、少なくともネットワークでソースサーバデバイス14 SERVER1、シンクサーバデバイス14 SERVER2とクライアントデバイス12を含むさらに他のA/Vモデルを示す図面である。前記クライアントデバイス12は、サーバデバイス14 SERVER1及びSERVER2と他のサーバデバイス14、例えばSERVER3及びSERVER4(図示せず)を選択し制御するために、使用者のための選択情報をディスプレイする使用者インタフェースのある管理者36を含む。選択情報はサーバデバイス14 SERVER1、SERVER2、SERVER3及びSERVER4を各々選択する使用者のためにセッション管理者36でServ1、Serv2、Serv3及びServ4に割当てられたアイコンシンボルを含む。前記ソースサーバデバイス14 SERVER1はDVCRを含むことができ、前記シンクサーバデバイス14 SERVER2は1/2DTVを含むことができる。

## 【0031】

動作の一例として、サーバデバイス14 SERVER1とSERVER2の選択において

、前記クライアントデバイス12は各サーバデバイス14のGC022をクライアントデバイスに伝送し各サーバデバイス14 SERVER1及びSERVER2に対応するGUI18をディスプレイする。使用者は、サービスのために対応するサーバデバイス14に制御と命令を提供する各サーバデバイス14のGUI18と相互作用できる。各サーバデバイス14は単独または他のサーバデバイス14と組合してサービスを提供できる。さらに、前記セッション管理者36は、サービスを行うために対応するサーバデバイス14が必要とするものであって、前記クライアントデバイス12で前記サーバデバイス14のGUI18間に制御状態データ26を伝送する。使用者制御と命令情報に基づいて、二つ以上のサーバデバイス14は使用者が求めるサービスを提供するためにそれら間で命令及び制御情報を通信できる。

### 【0032】

前記セッション管理者36は、ネットワーク10で多様なサーバデバイス14により提供される有用なホームネットワークサービスをアクセスしディスプレイする機能を有するソフトウェアエージェントを含むことができる。前記ソフトウェアエージェントは付加的にネットワーク10で多様なサーバデバイス14の性能を整合させ、互換性能を有する前記サーバデバイス14だけのための選択情報をディスプレイする。さらに、セッション管理者36は、使用者がサーバデバイス14に意味のある命令及び制御情報を提供するように一つのサーバデバイス14のGUI18で作られた選択と、他のサーバデバイス18のGUI18で作られた選択を整合させられる。

### 【0033】

動作のさらに他の例で、セッション管理者36はネットワークを検索し、ネットワークに接続されたサーバデバイス14を見つけるソフトウェアエージェントを実行する。前記ソフトウェアエージェントは、またサーバデバイス14の性能を決定するために各サーバデバイス14に貯蔵された性能データをアクセスし、使用者にその性能に対する情報を提供する。それから前記セッション管理者36は、図9に示したようにサーバデバイスSERVER1、SERVER2、SERVER3及びSERVER4のための選択アイコンServ1、Serv2、Serv3及びServ4をディスプレイす

る。

#### 【0034】

前記セッション管理者36は、使用者が4個の全てのアイコン中で選択できるように初期に全ての選択アイコンServ1、Serv2、Serv3及びServ4をイネーブルさせる。使用者がServ1選択アイコン上でクリックしてサーバデバイスSERVER1を選択した後に、前記セッション管理者36はサーバデバイスSERVER3及びSERVER4がサーバデバイスSERVER1と性能上互換できないと決定する。従って、前記セッション管理者36はサーバデバイスSERVER3及びSERVER4のために選択アイコンServ3及びServ4を各々ディセーブルさせる。それから使用者はサーバデバイスSERVER2を命令及び制御するためにアイコンServ2上でクリックできる。

#### 【0035】

使用者が選択されたサーバデバイス14のGUI18と相互作用することのように、使用者により各GUI18に入力された制御及び命令情報は使用者により後続するサーバデバイス選択に影響を及ぼす付加的な性能情報を提供する。例えば、もしVCRサーバデバイス14が選択されれば、他のサーバデバイス14のための選択アイコンをイネーブルまたはディセーブルさせることにおいて、セッション管理者36による後続の操作が再生または録画する使用者の決定により影響される。

#### 【0036】

ネットワークで各サーバデバイス14は、図9に示したサーバデバイスに関連した例を通じて前述されたように一つまたはそれ以上のサービス性能を有する。各サービス性能は情報のソーシングまたはシンキングを含む。例えば、TVはビデオ及びオーディオストリームを受信するシンキング性能を有し、VCRはビデオ及びオーディオ信号をソース(伝送)及びシンク(受信)でき、PCはビデオ、オーディオそしてデータを伝送及び受信できる。各ソーシング性能は相補的で互換的なシンキング性能を有する。これと類似に、各シンキング性能は相補的で互換的なソーシング性能を有する。例えば一つのデバイスでビデオ出力性能はさらに他のデバイスのビデオ入力性能により補完される。

## 【0037】

各デバイス14はネットワーク上でいろいろな他のサービスのためにソースまたはシンクされるので、図10に示したように各デバイス14は性能データテーブル(性能テーブル1)を貯蔵する。テーブル1の第1列はデバイス14のサービス性能を確認し、第2列は第1列に対応するサービスのために前記デバイス14がソースかシンクかをを確認する。新たなサービスはさらに古いデバイスと互換性を維持する間に性能データテーブル1を使用して実行される。例えば、もしさらに古いサービスと互換される新たなサービスが開発されるならば、新たなサービス及び古いサービス両方とも実行デバイス14は古いサービスを使用する古いデバイスと互換性を維持するにつれて、前記新たなサービスを行うデバイスのための性能データテーブル1に書き込まれることができる。

## 【0038】

第1実施形態で、デバイス管理者はソース及びシンクサービスデバイスのマッチングまたは比較を行う。例えば、前記デバイス管理者は互換性マッチングと共に多様なデバイス14及び指定されたデバイス14の性能及び特性を比較するソフトウェアエージェントの役割も行える。例えば、サービスがネットワークにわたって第1デバイス14から第2デバイス12までメディアストリームの場合、前記デバイス管理者は使用者が第1デバイス14の性能と互換される第2デバイス14を分別力あるように選択できるように第1及び第2デバイス14の性能を比較する。次はサーバデバイス14の実施形態のためのサービス性能リストの例を示す。

```
Stream_format_video_dv  
Stream_format_video_mpeg2tpt  
Stream_format_video_dsstpt  
Stream_format_video_mpeg2pes  
Stream_format_video_mpeg210901-tpt
```

## 【0039】

図11に示したように、各デバイス14はデバイスに関する特性を含む特性データテーブル(特性テーブル2)をさらに貯蔵する。名称と値段はテーブル2で各

特性を定義する。文字長さがテーブル2に示されているが、必要ではない。前記特性データはネットワーク10上で情報処理相互運用を容易にしデバイス情報を貯蔵する他のデバイス14に有用である。例えば、後述されるようにデバイスページはデバイス名称を貯蔵する特性テーブル2を使用する。他のフィールドは必要な時特性データテーブル2に追加されることができる。

#### 【0040】

前述された使用者—クライアントデバイス制御モデルで、特性データはクライアントデバイス12でサーバデバイス14のGUIページ上にディスプレイされる。他には、前記特性データをディスプレイするために第2レベルデバイス情報ホームページが使われる場合がある。さらに、テキストまたはXML(Extensible Markup Language)ファイルの形態の特性データがソフトウェアエージェントによりアクセスされる場合がある。デバイス—デバイス制御モデルのために、制御されたデバイスのための特性データはデバイスインタフェース及びアプリケーションインタフェースに貯蔵される。

#### 【0041】

特性テーブル2で前記デバイス位置特性フィールドは各デバイス14のための位置またはグループを貯蔵するために使われる。デバイスタイプ特性フィールドはVCR、DVD、DTV、カムコーダ、PC、保安システムのような特定のデバイス14のためのデバイスタイプを書き込む。前記デバイスタイプ特性フィールドは、もしデバイス自体がデバイスアイコンを供給しないならば、デバイスページでデバイスを表現するデフォルトデバイスアイコンを選択するために使われる。特性テーブル2はデフォルトソース及びデフォルトシンク特性フィールドのための多重の書き込みを含むことができる。そのようなそれぞれの書き込みは、デバイス14により処理される各データタイプのための他のデフォルトソースまたはシンクデバイス14を表現する。

#### 【0042】

望ましくは、性能及び特性データは階層言語を使用して構造化されたデータでパッケージされる。これはGCO伝送及びサーバデバイス—サーバデバイス制御のような他の目的で使われた性能と特性を回復させる共通方法を提供する。例えば

、特性データは次の構造化されたデータフォーマットを含むことができる。

```
<DEVICEATTRIBUTES>
  <ATTRIBUTE name=DeviceManufacturer value="Samsung Inc.">
  <ATTRIBUTE name=ManufacturerURL value=www.Samsung.com>
  <ATTRIBUTE name=ManufacturerIcon value="logo.gif">
  <ATTRIBUTE name=DeviceName value="Samsung DSS">
  <ATTRIBUTE name=DeviceModel value="SCH1900">
  <ATTRIBUTE name=DeviceType value=DDS>
  <ATTRIBUTE name=DeviceLocation value="Livingroom">
  <ATTRIBUTE name=DeviceIcon value="device.gif">
  <ATTRIBUTE name=DeviceAddress value=105.144.30.17>
</DEVICEATTRIBUTES>
```

#### 【0043】

一例として、性能データは次の構造化されたフォーマットを含むことができる。

```
<DEVICECAPABILITIES>
  <CAPABILITY type=MPEG2 value=Source>
  <CAPABILITY type=MPEG2 value=Sink>
  <CAPABILITY type=MPEG3 value=Source>
  <CAPABILITY type=MPEG3 value=Sink>
</DEVICECAPABILITIES>
```

#### 【0044】

アプリケーションインタフェース言語は、他のサーバデバイス13がサーバデバイス-サーバデバイスを含んでデバイス-デバイス制御を行うように使われる。前記アプリケーションインタフェース言語は命令語を含み、後述されたようにXMLを使用して記述されることができる。一つのサーバデバイス14の制御プログラム20は、GUI18を使用しないとまたは使用者の介入なしにネットワークを通じてさらに他のサーバデバイス14の制御プログラム20を遠隔で制御する。デバイス-デバイスの制御の例は自動操作である。使用者は初期に希望のサ

ービスのためにクライアントデバイス12を通した制御を提供し、次いで二つ以上のサーバデバイス14がサービスを提供するために使用者の相互作用なしでもお互い自動で通信し制御する。

#### 【0045】

図12及び図13を参照すれば、望ましくは標準アプリケーションインタフェース言語が、多様なサーバデバイス14内の多様な制御プログラム20間で情報処理相互運用を許容するように使われる。一実施形態で、前記標準アプリケーションインタフェース言語は、次のような構成ブロックの(1) サービス機能データベースのようなサービスの機能的規格ブロック40、(2) メッセージ要素が構成されるブロック(42)、(3) 産業標準フォーマットブロック44、(4) メッセージ圧縮ブロック46及び構造化されたメッセージデータを出力するメッセージ文字列構成ブロック48を含む。

#### 【0046】

図12は、命令メッセージ発生機能を行う構成ブロックの構成例を示す図面である。各メッセージアイテムはサービスの機能的規格で構成され、メッセージアイテムに対して産業標準圧縮形態(Hex)ラベルを選択することによって標準化される。メッセージアイテムのようなグループは完全な命令文字列を作るために組合せられる。CAL及びAV/Cのように存在する命令語は図12に示したように動作する。しかし、その命令語メカニズムは物理的なインタフェース上で物理的装置に対して2進コードまたは16進コードメッセージ及びシステム作動を規定しハードウェア規定に基づく。従って、その命令語は、制御器デバイス14内の一つのソフトウェアアプリケーションプログラム20がネットワーク上の制御されたデバイス14内のさらに他のソフトウェアアプリケーション20の位置を指定し制御するアプリケーションレベル、即ち、ソフトウェアレベルで、名称、アドレス、デバイス性能発見、通信言語そして命令メッセージを含む制御システム規格の制御メカニズムに基づいたネットワーク階層に対してあまり望ましくはない。前記制御メカニズムはコンピュータのような多用途、多重アプリケーションデバイスだけでなくDVCRのような装置を含むデジタル装置のようなデバイスに対してさらに適している。

## 【0047】

図13は、命令メッセージ発生機能を行う図12の構成ブロックの望ましい例を示す図面である。図13で、産業標準化フォーマット44とメッセージ圧縮46の位置が図12と違う。多数のテキスト標準化様式は完全なメッセージを作る機能的規格サービス40から選択される。後で前記メッセージはプロトコルスタックの下部階層により圧縮されたりする。図13は、電化製品CEに対してサービスまたはデバイス命令及び制御を行う方法を表す。メッセージ構成はXML標準化シンタックスにより定義され、メッセージ圧縮はHTTPのようなさらに他のプロトコル階層により行われる。命令インタフェース言語は下位ハードウェアレベルよりはアプリケーションソフトウェア20インタフェースレベルで使われる。従って、ネットワークプロトコルスタックは前記言語内の命令により管理され、各制御デバイス14と制御されたデバイス14はこれら間のメッセージ伝送のためにネットワークの統合された構成として示される。

## 【0048】

図14を説明すれば、クライアントデバイス12とサーバデバイス14との間の三つの相互作用の三つの実施形態を示す。第1実施形態“A”で、使用者は遠隔サービスアプリケーション“S”と通信する。前記使用者は使用者インタフェースとしてクライアントデバイスでブラウザを使用するが、ここで前記ブラウザはサービスアプリケーション“S”でサービスプログラム20を制御しHTML(Hyper Text Markup Language)またはXMLフォーマットで応答を受信する。XML基盤の非同期命令メッセージポストイングを受け入れるブラウザを備えた第二サーバが含まれる。例えば、DVCRにおいて第二サーバ14は“VCR FAILED:TAPE BROKE”のような命令メッセージを受け入れる。ブラウザを含むソフトウェアエージェントは使用者による後続アテンションとDVCRの制御のためにブラウザのGUI内で使用者のための命令メッセージがディスプレイするために用いられる。望ましくは、XML基盤のクライアントデバイス12は、サーバデバイス対サーバデバイス制御と命令のために他所で初期化された命令にตอบสนองするHTTP1.1サーバ性能を含む。

## 【0049】

第2実施形態“B”で、使用者はソフトウェアクライアント制御プログラム50

により取り替えられる。前記ソフトウェアクライアント制御プログラム50はXML基盤のサービスアプリケーション“S”への命令ポスティングを生成し、返信されるXML命令ポスティングを受信する。そして、第3実施形態“C”で、前記ソフトウェアクライアント制御プログラム50は、命令と応答が二つのサービスアプリケーション20間で交換されるサーバデバイス制御プログラム20のようなアプリケーションにより取り替えられる。この点において、実施形態“B”はナル(Null)サービスを有する実施形態“C”の特別な場合である。

#### 【0050】

XMLに基づいたアプリケーションインタフェース言語は、ワールドワイドウェブあるいはインターネットによりイネーブルされるデバイスまたはサービスのために第1サーバデバイス14と第2サーバデバイス14(デバイス-対-デバイスまたはサービス-対-サービス)間を制御するために使われる。アプリケーションインタフェース言語はウェブ標準、ミドルウェア階層に基づく。一実施形態で、デバイス-デバイス制御はネットワーク10でさらに他のサーバデバイス14から一つのサーバデバイス14で制御プログラム20またはアプリケーションを遠隔で制御することを含む。従って、アプリケーション20に対するインタフェース(API)はAPI拡張を使用するネットワークを通じて利用できる。望ましくは、前記API拡張は全体の情報処理相互運用を提供するためにXMLに基づいたインタフェースのような標準フォーマットを使用する。

#### 【0051】

なお、図15は、ネットワーク内で通信するサービスAとしてデザインされた第1アプリケーションAとBとしてデザインされた第2アプリケーションBのためのAPI拡張を示すブロック図である。例えばサービスAはネットワークで第1サーバデバイスAのための制御プログラムであり、サービスBはネットワークで第2サーバデバイスBのための制御プログラムでありうる。サーバデバイスBはサーバデバイスAに命令を伝送する。例えば、第1及び第2サービスデバイスA及びBはCEデバイスを含む。

#### 【0052】

サービスAのためのAPI拡張を参照すれば、第1最上位ブロック52は、CEデバ

イスを説明する英単語を使用してCE対象及び方法の総合的な定義またはデータベースを提供する。前記総合的な定義またはデータベースはC、XMLまたは対象及びそれら各自の方法を表現できる他のフォーマットでも存在できる。XMLを使用する総合的な定義またはデータベースはXCE定義と呼ばれる。第2ブロック54は、インタフェースデータタイプ定義INTERFACE.DTDでデザインされた全てのデバイス14のためにXML形式でAPIを表現するフォーマットを提供する。

#### 【0053】

手段AでデザインされたソフトウェアエージェントはサービスAのためのXCE定義のサブセットを使用し、XML形態文書を生成するサービスAのためのインタフェースデータタイプINTERFACE.DTD、INTERFACE-A.XMLを使用する。前記文書INTERF ACE-A.XMLはサービスAのための文書タイプ定義INTERFACE.DTDに従ってサービスAにより支援される対象及び方法を説明する。他のデータタイプ定義はまたINTERF ACE-A.XML文書を作成するために使われる。

#### 【0054】

ソフトウェア手段Aはまたネットワークインタフェース上で、サービスBから例えばCでプログラムされたサービスAまでXMLメッセージから変換されたルックアップテーブル56及び実行できる2進数にコンパイルされたルックアップテーブル56を生成する。望ましくは、ルックアップテーブル56はコンパイル時に作成され、ここでランタイムの間、サービスBから入力されるXML形態の方法(Method)メッセージ(命令)がサービスAのためにコンパイルされたアプリケーションコードにより作成されたAPI形式に変換される。前記ルックアップテーブル56はXML対象方法要請サービスとしてAのための装置固有の要請に変換するランタイム翻訳を提供する。ルックアップテーブル56はサービスAのためにサーバデバイスA上で局所的な実行のためのデバイス制御プログラム20と共にコンパイルされる。

#### 【0055】

INTERFACE-A.XMLは受信されたメッセージでエラーが発生した時正当性チェックのためにサービスAにより使われたりする。INTERFACE-A.XMLはまたサービスAと通信する前にサービスAのためのメッセージフォーマットを決定するサービスB

のような外部アプリケーションにより使われたりする。さらに、もしサービスBからサービスAへのメッセージがエラーを発生させれば、サービスBはエラーを診断するためにINTERFACE-A XML文書をアクセスできる。

#### 【0056】

サービスBに対するAPI拡張を参照すれば、前記第1ブロック58は前記のサービスAのためにXCE定義のようなCE対象の総合的な定義またはデータベースを提供する。次のブロック60はサービスAのためのAPIのような遠隔のAPIサービスまたはデバイスにXML形態方法(命令)呼び出しのための言語定義を提供する。前記言語定義はネットワーク上で対象との相互作用を説明する文書タイプ定義方法要請CALL DTDである。

#### 【0057】

手段Bでデザインされたソフトウェアエージェントは、命令をサービスBのためにコンパイルされたCプログラムコードをXML形式の方法要請に変換するルックアップテーブルを作成するために、少なくともサービスBのためのXCE規定内の対象及び方法のサブセットを用いる。従って、例えばルックアップテーブル62はサービスB(例えば、“再生”)により引き起こされた方法とサービスAでネットワークインタフェースを横切って方法呼び出しを行うXML文書またはメッセージ間で変換を提供する。ソフトウェア手段Bにより用いられるXCE定義のサブセットはネットワークの使用拡張及び本質に依存する。例えば、前記サブセットは、ホームネットワーク上で全ての可能なサービスの包括的あるいは制限された使用を提供するために選択されることができる。

#### 【0058】

従って、前記API拡張はXMLを使用するネットワーク上で多様なデバイス間で通信を提供する。前記の例で、サービスBのためのプログラムコード20はAPIで方法呼び出しを生成し、API呼び出しは内部デバイス通信のためのウェブ/インターネット標準XMLに適合にXML形態に変換される。XML方法呼び出し(メッセージ)はネットワークを通じてサービスAに伝送され、サービスAはネットワークインタフェースからのXML方法呼び出しをサービスAのためのプログラムコードAPI定義に再変換する。前記変換及び再変換は、他のデバイス間で2進互換性を異に要求す

るプログラムコードAPIを有するネットワーク内の多様なデバイスのためのウェブ/インターネット互換性を提供する。図15でのブロック図を使用するXMLインタフェースブロックの実施形態が下に示される。

---

```
interface.dtd
```

```
rules for describing an object interface in xml
```

---

```
<!ELEMENT parameter #PCDATA>
```

```
<!ATTLIST parameter valueCDATA #REQUIRED>
```

```
<!ELEMENT method (#PCDATA, (parameter)+)>
```

```
<!ELEMENT object (#PCDATA)
```

```
<!ELEMENT call (object, (method)+)>
```

---

```
interface.h
```

```
example object interface in c
```

---

```
/*object*/
```

```
typedef struct stream{
```

```
    int id;
```

```
};
```

```
/*method*/
```

```
void StreamPlay(int id, int speed);
```

```
void StreamStop(int id);
```

---

```
interface.xml
```

```
the same object in xml using rules of interface.dtd
```

---

```
<object>Stream
```

```
<method>Play
```

```

        <parameter type="int">id</parameter>
        <parameter type="int">speed</parameter>

    </method>
    <method>Stop
        <parameter type="int">id</parameter>

    </method>
</object>

```

---

call.dtd

rules for describing a c function call in xml

---

```

<!ELEMENT   parameter   #PCDATA>
<!ATTLIST   parameter   TypeCDATA   #REQUIRED>
<!ELEMENT   method       (#PCDATA, (parameter)+)>
<!ELEMENT   method       (#PCDATA, (method)+)>

```

---

controller.c

example controller command in c

---

```

.
.
.
StreamPlay(0xlae,500);
.
.
.

```

---

call.xml

the same command in xml using call.dtd

---

```

<!--example to play a stream-->
  <call>

      <object>stream</object>

      <method>Play</method>

      <parameter value="500">speed</parameter>

  </call>

```

### 【0059】

さらに、上記では、前述された利用可能なサービスの説明文書のINTERFACE.XMLの説明文書を作成するように使われたインタフェース定義INTERFACE.DTD及びCALL.DTDの例を示す。前記CALL.DTD定義は、XMLRPC(Remote Procedure Call)またはXMLRPCメッセージのような方法呼び出しまたは関数呼び出しメッセージを生成するための規定セットを含む。前記CALL.DTD定義は制御器サービス14の出力インタフェースを説明する。ホームネットワークで、例えばINTERFACE.XMLはホームネットワーク上に利用可能なサービスを表現する。利用可能なサービスはCE領域で全体サービスのサブセットである。

### 【0060】

OTR(One-Touch-Record)概要で、使用者は衛星STBのようなチューナーアクセスデバイスの制御を行う。使用者はプログラムリストを示すグラフィック的な使用者インタフェース表現のようなEPG(Electronic Program Guide)を使用してチューニングを制御する。OTR録画は、使用者が時間遅延された録画のためにVCRをプログラムするためにVCRグラフィック使用者インタフェースをアクセスせずに録画できるようにEPGから将来のプログラムの選択を含むサービスを提供する。OTRはVCRの制御を自動化する。下記はOPR.XMLの例示的な動作制御リストである：

(1) StreamOpen＝衛星STBからネットワークに出力された選択されたプログラムストリームを再生；OTRのためにこの制御はSTBデバイスに局部的である；(2) StorageOpen＝貯蔵サービスオープン；及び(3) StorageRecord＝ネットワークを通じてVCRに録画命令伝送。

---

call.dtd

rules for describing a c function call in xml

---

```

<!ELEMENT   parameter   #PCDATA>
<!ATTLIST   parameter   valueCDATA   #REQUIRED>
<!ELEMENT   method      (#PCDATA, (parameter)+)>
<!ELEMENT   object      (#PCDATA)>
<!ELEMENT   call        (object,method)>

```

---

interface.dtd

example for describing an object interface in xml

---

```

<!ELEMENT   parameter   #PCDATA>
<!ATTLIST   parameter   valueCDATA   #REQUIRED>
<!ELEMENT   method      (#PCDATA, (parameter)+)>
<!ELEMENT   object      (#PCDATA,method+)>

```

---

interface.xml

this document describes various CE services offered-a subset of the whole CE space.

---

```

<?xml version="1.0"?>
<!DOCTYPE interface SYSTEM "interface.dtd">
<object>Stream
    <method>Open
        <parameter type="int">id</parameter>
        <parameter type="int">channel</parameter>
    </method>
    <method>Close

```

```
<parameter type="int">id</parameter>

</method>

</object>

<object>Control

    <method>Set

        <parameter type="int">id</parameter>

        <parameter type="int">level</parameter>

    </method>

</object>

<object>Storage

    <method>Open

        <parameter type="int">id</parameter>

        <parameter type="int">channel</parameter>

    </method>

    <method>Record

        <parameter type="int">id</parameter>

    </method>

    <method>Play

        <parameter type="int">id</parameter>

        <parameter type="int">speed</parameter>

    </method>

    <method>Stop

        <parameter type="int">id</parameter>

    </method>

    <method>Close

        <parameter type="int">id</parameter>

    </method>

</object>

<object>Display
```

```

<method>Open
    <parameter type="int">id</parameter>
    <parameter type="int">channel</parameter>
</method>
<method>Render
    <parameter type="int">id</parameter>
</method>
<method>Blank
    <parameter type="int">id</parameter>
</method>
<method>Control
    <parameter type="int">id</parameter>
    <parameter type="int">cid</parameter>
    <parameter type="int">level</parameter>
</method>
<method>Close
    <parameter type="int">id</parameter>
</method>
</object>
<!-------
otr.xml
an xml representation of one touch record
c representation:
StreamOpen(100,2);/*play a stream(pushed by satellite feed*/
StorageOpen(24,2);/*open a storage service*/
StorageRecord(24);/*record the stream*/
-----
<?xml version="1.0"?>
<!DOCTYPE interface SYSTEM"call.dtd">

```

```

<call>
  <object>stream</object>
    <method>open</method>
      <parameter value="100">id</parameter>
      <parameter value="2">channel</parameter>
    </call>
  <call>
    <object>storage</object>
      <method>open</method>
        <parameter value="100">id</parameter>
        <parameter value="2">channel</parameter>
      </call>
    <call>
      <object>storage</object>
        <method>Record</method>
          <parameter value="100">id</parameter>
        </call>

```

#### 【0061】

図15に関して上で議論されたように、第1デバイスBはAのデバイス性能とAPIインタフェースの細部事項を調べるために第2デバイスAのINTERFACE.XML文書にアクセスし、第2デバイスAの支持された機能及び命令項目を決定できる。特に、前記第1デバイスBは第1デバイスBと第2デバイスAにより支援されたオーバーラッピング、そして従って利用できる方法を決定できる。図16は、アプリケーションBを含む第1サーバデバイスBがアプリケーションAを含む第2サーバデバイスAのINTERFACE-A.XML文書にアクセスする例を示す。前記第1サーバデバイスBは、第2サーバデバイスAでINTERFACE-A.XML文書と比較するためのINTERFACE-B.XMLを含む。

#### 【0062】

一つのシナリオで、前記第1サーバデバイスBはネットワークで第2サーバデ

デバイスAを制御しようとする。前記第2デバイスAのINTERFACE-A. XML文書は第2サーバデバイスAから第1サーバデバイスBに伝送され、第2サーバデバイスAの性能及びAPIインタフェース方法を質問するアプリケーションBにより使われる。これは第1サーバデバイスBがXML遠隔手順呼び出しXMLRPCを使用して第2サーバデバイスAを制御することを許容する。他のシナリオで、前記第1サーバデバイスBは少なくとも一回第2サーバデバイスAと通信を試みて通信確認が失敗した後、上記の段階を行う。また他のシナリオで、前記第1サーバデバイスBは第1サーバデバイスBにINTERFACE-A. XML文書を伝送せずに遠隔で第2サーバデバイスA内のINTERFACE-A. XMLに問い合わせる。

### 【0063】

INTERFACE-A. XML文書の項目を調べることに、前記第1サーバデバイスBは、前述されたようにXMLフォーマットで第2サーバデバイスAに伝送するための命令を創出できる。一般に前記第1サーバデバイスBは、前述されたように第1及び第2サーバデバイスB及びAにより使われたXCE定義サブセットでオーバーラップするINTERFACE-A. XMLの少なくとも一部項目を解釈できる。もし前記第1サーバデバイスBがINTERFACE-A. XML文書の一部項目を解釈できないならば、第1サーバデバイスBはその部分を無視したり、後述されるように翻訳によりその部分を説明するようにアシストするアプリケーションをフェッチできる。

### 【0064】

図17を参照すれば、制御器サーバデバイス14及び制御されたサーバデバイス14間のデバイス-デバイスまたは内部デバイスを制御する他の例が示される。前記制御器デバイス14は制御器アプリケーションを含み、制御されたデバイス14は実行可能なアプリケーションCを含む。制御されたデバイス14はINTERFACE-A. XML、アプリケーションCのアプリケーションインタフェース説明をさらに含む。アプリケーションEは、制御されたデバイス14の性能及び制御されたサーバデバイス14のAPIインタフェース方法を問い合わせるために制御されたデバイス14内のアプリケーションインタフェース説明Aをアクセスする。次に、アプリケーションEは、制御されたデバイス14のハードウェアまたはサービスDを制御するためにXML遠隔手順呼び出しを使用してアプリケーションCを命令

し制御する。スケジューラデバイスはVCRの時間一遅延一録画制御器のように定められた日になれば駆動する制御器デバイス14の場合でありうる。

#### 【0065】

第1例で、前記アプリケーションEは、ネットワークを通じた遠隔照会によりアプリケーションインタフェース説明Aをアクセスする。第2例で、前記アプリケーションEは、制御されたデバイス14から制御器デバイス14にアプリケーションインタフェース説明Aのコピーを伝送することによってアプリケーションインタフェース説明Aをアクセスする。次に、アプリケーションEは局部的にインタフェース説明Aに問い合わせる。第3例で、アプリケーションインタフェース説明Aはインタフェース説明のためにライブラリ空間を提供するライブラリデバイス64に伝送され、アプリケーションEはライブラリでインタフェース説明Aを遠隔で問い合わせる。前記ライブラリデバイス64は直接的な制御動作及び応答を使用できる関連したアプリケーションのアドレス(URL)を貯蔵する。

#### 【0066】

図18を参照すれば、XMLプロトコルはネットワークで多様なデバイス14のアプリケーション20間のAPIレベルで通信スタック66においてウェブ標準共通ミドルウェア階層を提供する。各デバイス14で通信スタックの最上部アプリケーションはネットワークを通じて通信メッセージを送受信し、デバイスのためのデバイスハードウェアまたはサービスソフトウェアを局部的に制御するデバイススタックでソフトウェア階層と通信する。

#### 【0067】

XML層OUT 68でデザインされた第1 XML層APIはメッセージを送信するために使われ、XML層IN 70でデザインされた第2 XML層APIはメッセージを受信するために使われる。XCE定義と方法呼び出しのXML定義、即ち前述された文書タイプ定義CALL. DTDは、XML層OUT 68を生成するように使われる。さらにXCE定義及び方法呼び出しのためのXML定義、即ち前述された文書タイプ定義INTERFACE. DTDは、XML層IN 70を生成するように使われる。例えば制御器アプリケーションはXML層OUT 68を使用し、制御されたアプリケーションENXML層IN 70を使用する。

#### 【0068】

図19を参照すれば、サーバデバイス—サーバデバイス命令及び制御構造のさらに他の実施形態を示す。XML—基盤制御構造は、ウェブとインターネットにより駆動されるデバイスまたはサービスのためのデバイス—デバイス(サービス—サービス)のために使われる。第1デバイスAは、XML命令メッセージを使用するネットワークを通じて第2デバイスBでアプリケーション20を遠隔で制御できる。各デバイスに対するインタフェースは前記デバイス内でアプリケーションに対するインタフェースを含みXMLフォーマットで述べられる。後述されるように、前記インタフェースはネットワークを通じて他のデバイスによる訂正及び説明のためにミドルウェア階層上で拡張され使用可能になりうる。

#### 【0069】

それぞれのサーバデバイスA及びBはネットワークを通じて他のサーバデバイスを制御し、ネットワークを通じて他のサーバデバイスにより制御されるためのハードウェアとソフトウェアを含む。図19で、ホームネットワークデバイスAは制御器デバイスまたはモジュールであり、ホームネットワークデバイスBは制御されたデバイスまたはモジュールである。それぞれのデバイスA及びBは、インタフェース文書INTERFACE.XML及び文書タイプ定義INTERFACE.DTDで構成された局所的なデバイスXMLインタフェース72を含む。INTERFACE.XML文書は対応するデバイス14により支援される対象、方法及びパラメータの説明を含む。前記INTERFACE.DTD文書は、前述されたようにデバイスのXMLインタフェースで具体的な正当性チェックのために使われたりする。

#### 【0070】

それぞれのデバイスA及びBはまたXMLインタフェース及びXMLRPC命令のようなXMLメッセージをパッシングし確認するためのプログラムコードで構成されたXMLパーサ(Parser)74を含む。前記XMLパーサ74は図18に関して前述された前記XML階層IN70と類似である。さらに、それぞれのデバイスA及びBは、XMLRPCの方法名とパラメータを再生するためにXMLRPCメッセージから送出する方法名とパラメータをエンコーディングするためのXMLRPCエンコダ及び入力されるXMLRPCメッセージをデコーディングするためのXMLRPCデコダを含む。XMLRPCコーデック76はデバイスXMLインタフェース72及びデバイス—デバイス制御構造か

ら独立され、それよりデバイス—デバイス制御構造の他の特徴を変化させずに他のXMLRPCフォーマットの使用を許容する。

#### 【0071】

プログラムコードで構成されたインタフェースフェッチャ(Fetcher)は、さらに他のデバイスまたはホームネットワークインタフェースライブラリ80から直接的にさらに他のデバイスのデバイスインタフェースをフェッチするためにそれぞれのデバイスA及びBにより使われる。デバイス14が制御器デバイスであれば、制御器デバイス14内の制御器アプリケーションプログラムコード82は、XMLパーサ74、インタフェースフェッチャ78及びXMLRPCコーデック76のような制御器デバイス14内のソフトウェア及びハードウェアを統制することによって、ネットワークを通じて他のデバイス14の命令及び制御に影響を与える。デバイス14が制御されたデバイスであれば、制御されたデバイス14内の制御されたアプリケーションプログラムコード84は他のデバイス14により制御されるデバイス14のためにデバイス14内のソフトウェア及びハードウェアを統制する。それぞれのデバイスA及びB内のホームネットワークデバイスウェブサーバ86はネットワーク上のデバイス間の通信を管理する。各デバイスA及びB内の固有ルックアップテーブル88に対するXMLはX、MLRPCメッセージ(例えば方法名称、構成要素名称及びタイプ)をデバイスの固有インタフェース(例えば固有方法名称、構成要素及びタイプ)に変換する制御されたアプリケーション84により使われる。前記テーブル88は、XMLメッセージ内で方法及びパラメータの名称とデバイスの固有インタフェースが同じ場合には使われない。

#### 【0072】

それぞれのデバイスA及びBは一つ以上のハンドラ(Handler)90をさらに含み、各ハンドラ90は制御されたアプリケーション84内部から特定デバイス機能の固有具現へのポインタを含む。大部分のデバイスでデバイス機能の固有具現は実行時間での2進コードを含む。2進コードは例えばC及びジャワを含んで、コンパイル時間に高水準の言語から生成される。従って、電化製品製造者は存在するハンドラ及び機能具現に影響を及ぼさずに新たな機能のためのより優れたハンドラ90を追加できる。それぞれのデバイスA及びBでハードウェアサービス92

はデバイス機能の固有具現を含む。それぞれのデバイスA及びBはまたデバイス機能の固有具現でAPIを具備する固有インタフェース94を含む。

### 【0073】

さらに、ホームネットワーク対象要請ブローカ(HNORB: Home Network Object Request Broker)79及びインタフェースライブラリ(IL: Interface Library)のようなネットワーク対象要請ブローカはホームネットワーク10のためのミドルウェア階層98を提供する。図19に示したように、前記ミドルウェア階層98は第3デバイス96または分離された制御ハブに位置できる。前記HNORB79は、ネットワーク10に接続された他のデバイス14の存在を見つける一つのデバイス14により使われるためのソフトウェアエージェントを含む。前記HNORBソフトウェアエージェントはデバイス名称を名称の階層的なツリー構造で組織し、デバイスインタフェースを前記検索できるインタフェースライブラリで組織し、そしてインタフェース情報を要請するデバイスにデバイスインタフェースを提供する。

### 【0074】

HNORB79及びIL80で構成されたミドルウェア階層は、指定された局所的なホームネットワーク10の外部からアクセスされる選択されたホームデバイスのようにインターネットに直接連結される。一つの局所的なホームネットワーク内のミドルウェア階層98は、二つのホームネットワーク10を備える統合されたネットワークを提供するためにインターネットを通じて他の局所的なホームネットワークでミドルウェア階層98に接続される。このような場合に、適当なストリーム符号化を有する認証された使用者はビデオを再生しTVでそれを見るために使用者の第2家庭内のTVから使用者の第1家庭内のDVDチェンジャ(Changer)にアクセスできる。

### 【0075】

前記インタフェースライブラリ80を使用するために、少なくとも一つのHNORB&ILは局所的なホームネットワーク10上で動作中であるべきである。一つ以上のHNORB&ILがまた利用される。例えば、ケーブルモデム、いくつかのDTV、そして中央家庭ハブは全てそれら自体のHNORB&ILソフトウェアエージェントを有する

ことができる。HNORB&ILの位置を定めるためにデバイス14は局所的なホームネットワークを通じて放送メッセージを伝送する。デバイス14で応答した第1HNORB&ILはデバイス14により使われる。一応HNORB&ILが位置すれば、デバイス14及びHNORB&ILは登録、インタフェース要請及びフェッチのための地点間TCP(Transmission Control Protocol)またはUDP(User Datagram Protocol)とデバイスルックアップサービスを確認できる。もしUDPプロトコルを利用できなければ、TCPプロトコルがIEEE1394のような高帯域幅接続のために使われる。HTTP-基盤のXMLRPCはHNORB&IL通信のためのデバイスのために使われる。例えば、デバイス14は一つ以上の構成要素としてデバイスインタフェースをパスするためにHNORBの“レジスタ”方法を遠隔で呼び出すことができ、またはXMLRPC呼び出しはXMLRPC応答または戻る値としてILから部分的または全体的なデバイスインタフェースを再生できる。

#### 【0076】

前述したように、一つ以上のHNORB&ILは同時に局所的なホームネットワーク10で動作でき、ここで各HNORB&ILは使用可能なデバイスのサブセットを認識でき、一つのHNORB&ILは見つけられないデバイス14を位置させるために他のHNORB&ILと通信できる。一つの局所的なホームネットワーク10上の多重HNORB&ILは、UDP及びTCPのような放送メッセージを使用することによって自動で相互間に位置できる。このような場合に、多重HNORB&ILは多数のインタフェースライブラリ80が分散インタフェースライブラリを構成する間分散対象要請ブローカを構成する。故障許容誤差を提供するために、もし一つのHNORB&ILが突然終了しなければならないならば、このHNORB&ILと共に登録された全てのデバイスは通知され、前記デバイスは他の有用なHNORB&ILと共に自動で登録される。

#### 【0077】

各デバイスインタフェースは組合された一貫的で単一の論理的な名称を有している。他のデバイスは、前記デバイスの位置または実際ネットワークアドレスが変わった後にもデバイスを認識しアクセスする前記一貫的で単一の論理的な名称を使用できる。論理的な名称のマッピングと実際デバイスアドレスのマッピングはHNORBで命名サービスのためのソフトウェアエージェントにより扱われる。望

ましくは、標準化された命名方法が使われる。より望ましくは、デバイス名称を階層的ツリー構造で操作するために階層的な命名構造が使われる。この階層的な構造はファイルシステムと類似に“/”を使用して表現できる。その構造は、家庭/MPEG 2/TVのような他のサービスタイプにより、または家庭/居間/VCRのような他の位置によるもののように、他の方法で生成される。いろいろな命名ツリーは実行と能率のために共存できる。

#### 【0078】

図19での制御器サーバデバイスA及び制御器サーバデバイスB間の命令及び制御例で、前記ミドルウェア階層98は第3デバイス96にあたり、または分離された中心ハブにありうる。暗いブロックは、図19に示した特殊な命令及び制御プロセッサのために使われたデバイス構成要素を示す。動作シナリオで、デバイスA及びBがネットワークを通じて使用可能になりアクセス可能になった後、各デバイスは中央HNORB及びILミドルウェア層98にそれ自体及びそのXMLインタフェースを書き込み/提出する。もし中央HNORB及びILミドルウェア階層が使用できなければ、各デバイスはそれ自体を知らせるために局所的なホームネットワークを通じてメッセージを放送する。

#### 【0079】

前記デバイスAの制御器アプリケーション82は制御されたデバイスBの全体または部分のデバイスインタフェースに問い合わせを試みる。もしインタフェースライブラリ80が有用でなければ、制御器デバイスAはネットワークを通じてデバイスBに要請を最初に伝送し、それからデバイスBからデバイスBのXMLインタフェースを受信することによって、制御器デバイスBから直接的に制御されたデバイスBのデバイスインタフェースを求めフェッチできる。しかし、もしインタフェースライブラリ80が使用できれば、制御器デバイスAはインタフェースライブラリ80から制御されたデバイスBの全体または部分のデバイスインタフェースを求められる。HNORBのソフトウェアエージェントはインタフェースライブラリ80から前記デバイスBのXMLデバイスインタフェースを得、制御器デバイスAにそれを回送する。

#### 【0080】

一応制御器デバイスAが制御されたデバイスBのXMLデバイスインタフェースを受信すれば、デバイスAの制御器アプリケーションはデバイスBのデバイスインタフェースを分析し翻訳するためにデバイスAのXMLパーサ74を使用する。それからデバイスAのXMLRPCコーデック76はパーシング結果を使用して望ましいXMLRPC命令メッセージを生成する。前記XMLRPC命令メッセージはネットワークを通じて制御されたデバイスBに伝送される。前記XMLRPC命令メッセージを受信すれば、デバイスBの制御されたアプリケーション84は受信されたXML命令メッセージを分析し翻訳するためにデバイスBのXMLパーサ74を用いる。それからデバイスBのXMLRPCコーデック76は、求められたサービスを行うためにデバイスB機能のための方法名称及びパラメータを含む命令メッセージから方法呼び出し情報を得るためにパーシング結果をデコーディングする。

#### 【0081】

デバイスBの制御されたアプリケーション84は、次にデバイスBの固有インタフェースを通じてデバイスBの固有機能実行をアクセスし送出するためにデバイスBで固有ルックアップテーブル88及びハンドラ90に対してXMLを使用する。もし関数がある応答または戻る値を生成すれば、前記応答または戻る値はXMLまたはXMLRPCメッセージにエンコーディングされ、制御器デバイスAに伝送される。さらに、ミドルウェア階層HNORB及びILは制御されたデバイスBに対する参照を制御されたデバイスAに提供し、ここでデバイスAは局所的なデバイスA固有機能に対する呼び出しと同じようにデバイスB固有機能に対する遠隔呼び出しを発生させる。

#### 【0082】

望ましくは、全てのデバイスがネットワークを通じてRPC呼び出しを説明しデコーディングできるように、標準XMLRPCフォーマットが使われる。制御されたデバイス14のデバイスインタフェースが制御器デバイス14により問い合わせられ調査されるために、望ましくは能率を向上させるために充分なデバイスインタフェース情報を有する単純化されたXMLRPCフォーマットが使われる。次の例はOTR(One Touch Record)及びTDR(Time Delayed Record)動作のためにXMLRPCを呼び出す二つの可能なフォーマットを示す。

## 【0083】

例1:

XMLRPC呼び出し、詳細タグ及びインタフェース情報を含むフォーマット例:

1. OTR呼び出し例:

```
<?xml version="1.0"?>
<call>
  <object>DVCR1.record</object>
  <method>timeDelayedRecord</method>
  <parameter>
    <parameter>
      <name>channel</name>
      <value><int>4</int></value>
    </parameter>
    <parameter>
      <name>recordTime</name>
      <value><time>2:10:30</time></value>
    </parameter>
  </parameters>
</call>
```

2. TDR呼び出し例:

```
<?xmlversion="1.0"?>
<call>
  <object>DVCR1.record</object>
  <method>timeDelayedRecord</method>
  <parameter>
    <name>channel</name>
    <value><channelName>NBC</channelName></value>
  </parameter>
  <parameter>
```

```

    <name>startTime</name>
    <value><datetime.iso8601>19990401T19:05:35<
      /datetime.iso8601></value>
  </parameter>
</parameter>

    <name>recordTime</name>
    <value><time>2:00:00</time></value>
  </parameter>
</parameters>
</call>
【0084】

```

#### 例II:

XMLRPC呼び出し、減ったタグ及びインタフェース情報を有するフォーマット例

:

#### 1. OTR呼び出し例:

```

<?xml version="1.0"?>
<call>
  <object>DVCR1.record</object>
  <method>timeDelayedRecord</method>
    <parameter value="4">channel</parameter>
    <parameter value="2:10:30">recordTime</parameter>
  </call>

```

#### 2. TDR呼び出し例:

```

<?xml version="1.0"?>
<call>
  <object>DVCR1.record</object>
  <method>oneTouchRecord</method>
    <parameter value="NBC">channel</parameter>
    <parameter

```

```

value="1990401T19:05:35">startTime</parameter>
<parameter value="2:00:00">recordTime</parameter>
</call>

```

#### 【0085】

図20を参照すれば、ホームデバイス14のためのデバイスインタフェースは、標準化された用語を使用する産業標準構造化されたデータベース100に基づく。新たなインタフェースのためのインタフェースデータ及び用語はデータベース100に追加される。CE対象を説明するために英単語を使用する包括的な定義あるいはデータベース、方法そしてパラメータはCEデータベース102に規定される。包括的な定義またはデータベースは対象及びそれらそれぞれの方法及び構成要素を示すC、XMLまたは他のフォーマットになりうる。標準化されたXML用語を使用する包括的な定義またはデータベースはXCE定義またはデータベース104と呼ばれる。

#### 【0086】

制御器及び制御されたアプリケーション82、84は、XCEデータベース104に基づいたXMLの標準インタフェースサブセットを使用してプログラムされる。各デバイスインタフェースはXML形態で前記アプリケーション82、84と共に貯蔵される。たとえ前記XCEデータベース104がXMLにある必要がないとしても、図15に関して前述されたように、本発明の実施形態でコンパイル時に作成された前記サブセットインタフェースはXML内にある。

#### 【0087】

図20で、埋め込まれた装置14のために'製造会社'情報としてデザインされた情報は製造時装置14に内蔵されたものであり、'ホームネットワーク'としてデザインされた情報はネットワークで装置の動作の実行時間特徴の一部である。Nデバイスのための1...NとしてデザインされたデバイスXMLインタフェース72は、標準化されたXCEデータベース104でデータの一部分である。ホームネットワークインタフェースライブラリ(HNIL: Home Network Interface Library)106はホームネットワークに接続された使用可能なデバイス14のデバイスインタフェース集合を提供する。HNIL 106は全体XCEデータベース104のサブ

セットである。

#### 【0088】

図16で、デバイスインタフェースはデバイスAのためのインタフェースの内容を調べるデバイスB内のアプリケーションBのためにデバイスAからデバイスBに伝送される。前述されたように、デバイスインタフェースはデバイスにより維持される対象、方法パラメータの説明を含み、例えばデバイスAに対するINTERFACE-A.XMLとして言及される。デバイスXMLインタフェース72はXMLフォーマットのデバイスインタフェースである。XCEデータベース104の内容はデバイスインタフェースを提供するサービス基盤の構造である。

#### 【0089】

図20を参照すれば、前記XCEデータベース104は、またCEデバイス14を表現するXMLを使用するために標準化された規定セットを提供するCEデバイスのための標準化されたXCEインタフェース文書タイプ定義(DTD: Document Type Definition)を提供する。前記DTDまたはそのサブセットは正当性チェックのために使われる。製造会社手段108としてデザインされたソフトウェアエージェントは、規定されたCEデバイスのための標準化されたXCE定義のサブセットをフィルタリングして使用するが、例えばINTERFACE.XML及びINTERFACE.DTDのようなCEデバイスのXMLデバイスインタフェース72を生成する標準化されたXCEインタフェースDTDを使用する。前記文書INTERFACE.XMLは標準化されたXCEインタフェースDTDに従って特殊なデバイスにより支援される対象、方法及びパラメータの説明を含む。前記文書INTERFACE.DTDは標準化されたXCEインタフェースDTDのサブセットであって、デバイスのXMLインタフェースのための正当性チェックのために使われる。他の文書タイプ定義もINTERFACE.XML文書を生成するために使われたりする。

#### 【0090】

前記XMLインタフェース文書及び前記DTD文書を含むCEデバイスのXMLインタフェース72は、ホームネットワークインタフェースライブラリ106のような一般的にアクセスできるライブラリに貯蔵される。ソフトウェアエージェント110はネットワーク上の全てのアクセスできるデバイス14のデバイスインタフェ

ース72を収集し、デバイス名称/アドレス情報と共に検索できる構造化されたインタフェースライブラリ106にそれらを配置する。前記インタフェースライブラリ106はXCEデータベース104のサブセットであって、インタフェースライブラリ106を作る過程は部分または全てのXCEデータベース104を再構成することと類似である。前記インタフェースライブラリ106は、ホームネットワーク内の全てのデバイスのデバイスインタフェースの集合または最も最近に使われたデバイスインタフェース72が貯蔵されるホームネットワークで全てのデバイス14の有用性に依存するキャッシュ(Cache)として機能できる。デバイス14が、DVD再生器でディスクを変えることのようなイベントに起因してそのデバイスインタフェース72を更新した場合には、デバイスインタフェース72の一部がイベントサービスに基づいて更新される。

#### 【0091】

図21を参照すれば、望ましくは各デバイス14のデバイスインタフェース定義72は階層的な形態を有する。なぜならこれはホームデバイスにおいて、デバイスインタフェース定義72が長くなる可能性があるからである。典型的に、時間遅延された録画のための単一機能のような一つまたは少数の機能は同時にアクセスされ、従って前記デバイスインタフェース72の唯一に小さな部分が使われる。全体デバイスインタフェース72を作成することよりはデバイスインタフェース72の一部分だけを作成することがより効果的である。階層的なデバイスXMLインタフェースを使用することによって、制御器デバイス14は、制御器デバイス14またはHNORB及びILミドルウェア階層98からXMLデバイスインタフェースのための要求において望ましい機能カテゴリーまたは機能を特定することによって、制御されたデバイス14の部分的なデバイスインタフェース72を求められる。後者の場合に、HNORB及びILミドルウェア階層98はデバイスインタフェース72の望ましい部分を回送する。

#### 【0092】

図21を参照すれば、前記階層的なデバイスインタフェース構造は、(1)現在有用なデバイスを書き込んだ各ホームネットワークのXMLインタフェースのための第1階層112、(2)機能カテゴリーを書き込んだ各デバイスの一般的なX

MLインタフェースのための第2階層114、(3) デバイスのための各機能カテゴリのXMLインタフェースを規定するための第3階層116、(4) 機能カテゴリ内の各機能のXMLインタフェースを規格化するための第4階層118を含む4つの層を含む。ホームネットワーク内部で、ただ三つの下位層114、116及び118が使われ、ホームネットワーク外部で第1層112が使われる。

### 【0093】

図22は、前記層112、114、116、118及び対応するインタフェース例を示す。各階層内のインタフェースは(できれば)両方向リンクを提供するXリンクまたはXポインタのようなリンクを通じて上位または下位層にリンクされる。Xリンクは(1) XML文書内のリンクをリンクとして認識させるXリンク成分、(2) リンクをXML文書の正確なサブパートにアドレッシングさせるXポインタ成分の二つのパートを有するハイパーリンク機能性のパッケージを含む。従って、XリンクはどのようにリンクがXML文書に挿入されるかを管理するが、ここで前記リンクはGIFファイルのようなデータを示す。さらにXポインタは、どこからでも(例えばHTMLファイルから)、XML文書にリンクする時URLに行ける断片識別子(Fragment Identifier)を決定する。

### 【0094】

本発明に係る他のサーバデバイス14を制御するサーバデバイス14のための典型的な命令及び制御モデルで、第1デバイス14は第2インタフェース階層114で第2デバイス14のデバイスインタフェースを問い合わせるように試みる。機能カテゴリ(FC: Function Category)を選択した後、前記第1デバイス14は、録画カテゴリのような第2デバイス14で特定の機能カテゴリのインタフェース階層116を問い合わせる。さらに、前記第1デバイス14は前記機能に対する呼び出しを作るためにOTRまたはTDRのように特定の機能をインタフェース階層118に問い合わせる。階層的またはツリー構造はより効果的なインタフェース機能を探索するようにしてネットワーク帯域幅を縮める。インタフェースファイル構造及び層の例が下記の通りである。

第1層112-HN1.xmlXML

第2層114-VCR1.xml

第3層 1 1 6 - VCRI\_RecordCategory.xml

第4層 1 1 8 - VCRI\_RecordCategory\_OTR.xml

#### 【0095】

類似に、ホームネットワークインタフェースライブラリ 1 0 6 は階層的であり、デバイスの他のサービスタイプまたはルームのような他の位置により多様な方法で構造化できる。前記階層的な構造は他のホームネットワークまたはインターネットに対する局所的なホームネットワーク 1 0 のインタフェースである。

#### 【0096】

XMLシンタックス (Syntax) で具現される例示的で階層的なデバイスインタフェース定義 7 2 が下に示される。

```
consumer(document_file, doc)
```

```
+----document_file<server_home.dtd, server_auto.dtd>
```

```
+----doc (services_home, server_auto, server_samsung_web_site,  
avc_commands, cal_commands,,)
```

```
+----services_home (xml_utility, client, server_av, lighting, comms,  
hvac, utility, security, appliances, convenience,,)
```

```
+----xml_utility (download_DTD_file,,)
```

```
+----client (acknowledge, attention, error, post_message, sound,  
stop_schedule, stop_all,,)
```

```
+----sound (alarm, ring, buzz,,)
```

```
+----server_av (controls_gen, source, sink)
```

```
+----controls_gen (ping, process_infor, setup,,)
```

```
+----process_info (s/w_id, h/w_id)
```

```
+----h/w_id (ser_no, manuf, model, class,,)
```

```
+----s/w_id (ser_no, exe_name, version,,)
```

```
+----setup (clock,,)
```

```
+----clock (hours, minutes, seconds)
```

```
+----source (service_id, media, rate, protocol, stream_format,  
controls_av,,)
```

```

+---sink(service_id,media,rate,protocol,stream_format,
controls_av,,)
+----service_id(url,,)
+----media(tpt_stream,ram,disk,tape,,)
+----disk(name,number,,)
+----rate<value>
+---protocol(61883/1394,UDP/IP/Ethernet,,)
+----61883/1394(isoch_ch_no)
+---stream_format(video、audio,,)
+----video(dv,mpeg2tpt,dsstpt,mpeg2pes,mpeg10801-tpt,)
+----audio(mpeg3,ac-3,midi,,)
+controls_av(flow_control,tune,timer_record,ui_control,,)
+----timer_record(tune,flow_control)
+----flow control(play,stop,goto,record,,)
+----play(time_params)
+----record(time_params)
+----time_params(now,start,duration,end,,)
+----tune(send_epg,channel,,)
+----channel(number,id,time_params,,)
+---ui_control(display,acoustic)
+----display(brightness,contrast,color/tint,
horiz_size,vert_size,,)
+----acoustic(volumn,base,treble,balance,fade,)
+----lighting(sensors,lights,send_epg)
+---sensors(living_room,sky,,)
+---lights(rooms_up,rooms_down,yard,,)
+----rooms_up.bed1,bed2,bed3,bed4,,)
+----bed1(lamp,dimmer,,)
+----dimmer<value>

```

```

+----rooms_down(family, kitchen, living, dining, soho, garage,,)
+-----yard(front, back)

+----comms(homehub, intercom, telco,)
+----homehub(send_device_list, send_configuration, sene_snmp_mib,,)
+----intercom()
+----telco()

+----hvac(controls_gen, controls_hvac,,)
+----controls_hvac(a/c, heat, temp, humidity,)
+----temp(low, high, hysteresis,,)
+----utility(meters, energy_mgmt,,)
+----meters(water, gas, electric,,)
+----water<value>, gas<value>, electric<value>
+----security(sensors, send_epg, alarm,,)
+----sensors(peripheral, motion,,)
+----peripheral(rooms_up, rooms_down,,)
+----motion(room_down, yard,,)

+----appliances(microwave, range, oven, fridge, freezer, coffee, toaster,
washer, dryer, water_heater,,)
+----microwave(send_epg, controls,,)
+----fridge(temp,)
+----water_heater(temp,)

+----convenience(window, curtain_open, door/gate, pool/spa, bath,
fountain, lift, jacuzzi,,)
+----curtain_open<value>
+----server_auto(message, server_auto_ford_explorer_98,,)
+----server_auto_ford_explorer_98(mileage, maintenance,,)
+----mileage<data>
+----maintenance<data>

+----server_samsung_web_site(message, service, help,,,)
```

```
+----avc_commands<,,,command_string,,,>
+----service_id(url,,)
+cal_commands<,,,command_string,,,>
+----service_id(url,,)
```

### 【0097】

前記階層的なデバイスインタフェース定義72は次のような分野を含む。

'document file'、これはXCEデータベース104またはXCEデータベース104のバージョン部分の正当性及び訂正の検証のためにXMLパーサにより使われる文書タイプの規定(DTD)ファイルの名称を提供する。前記XCE構造の相異なる部分のためのいろいろなDTDファイルが考えられ、ここで前記DTDはRPC CALLのための文書タイプ定義及び通信のためのINTERFACE DTDとは違う。

'doc'、これは互換性、特性、通信及び制御インタフェースの領域の最上位レベル名称を提供する。

'Services\_home'、これは家庭自動化、電化製品、効用性のための領域を提供する。

'Server\_auto'、これは車庫内にある自動車に対して、一台またはそれ以上の自動車タイプに対する有用なメッセージインタフェースを示す。例えば、'server\_auto\_ford\_explorer\_98'は特定自動車に対するインタフェースである。これは自動車のマイレージ及び保守インタフェースをアクセスできるようにし、直接チェックする自動車製造会社または車庫が直接チェックまたは遠隔診断する遠隔アクセスで使われる。

'server\_samsung\_web\_site'、これは家庭外部の製造会社ウェブサイトと通信し、メッセージ、サービス、アシストなどのためのインタフェースを含む。

'AVC\_commands'及び'CAL\_commands'、これは例えばAV/C及びCAL言語を解釈できる引き継ぐんだデバイスを提供する。この構造部分は前記言語で命令を確認し、命令はXMLで付加されたり行われる。従って、前記項目はXCE(Web)対象ではなく、プロトコル変換器アプリケーションは元のCALまたはAV/Cアプリケーションソフトウェアへのインタフェースとして使われる。

### 【0098】

前述したように、'Services\_home'はA/V電化製品を含むメーン構造を提供する。前記構造の一部分は特定な例としてビデオサービスシンク及びストリーム目的地(例えば、DVCR)制御インタフェースに対し詳細に拡張される。典型的なホームネットワークで制御インタフェースは次を含む：

'xml\_utility'、これは更新されたDTDファイル、インタフェースファイル、プログラムファイルをダウンロードすることのよう支援ユーティリティネットワーク機能を支援するための細部事項を提供する。

'client'、これはウェブブラウザを含むクライアントデバイス12のインタフェース細部事項を説明する。例えば、'acknowledgment'は發送されたメッセージまたは命令で制御器の認証承知を表示する。

'server\_av'、これはSTB、DVCR、DTV、DVD、AUDIOなどを含む有用な全てのオーディオ及びビデオサービスに対して制御及び容量インタフェースを提供する。

'lighting'、これは家庭自動化照明制御器にインタフェースを提供し、センサー及び照明などを含む。

'comms'、これは典型的に使用目的またはデバイスのセッティングまたは構成要素の遠隔管理または構成を復旧するために通信デバイスに制御インタフェースを提供する。

'hvac'、これはHVACシステムの遠隔制御のためのインタフェースを提供し、例えばその日のピーク負荷期間中に家庭のHVACシステムを消すために公益設備会社(Utility company)により家庭外部から前記システムを制御するために使われる。さらに、前記インタフェースは、自動温度調節装置制御よりさらに複雑な制御メカニズムを提供する制御器に基づいたデバイスのための装置により、家庭内部からHVACシステムを制御するために使われる場合もある。

'utility'、これは例えば、家庭のための計量器を讀取るためのインタフェースを提供する。

'security'、これは保安センサー及びアラームセッティングのためのインタフェースを提供する。従って、ホームネットワークデバイス上でインタフェース使用、アプリケーション進行はそのデバイスの監視及び制御のための家庭周辺のセンサー及び感知器をアクセスできる。

'appliances'、これは例えば、遠隔制御または温度設定感知または制御器デバイスから他の制御及び構成要素を含む台所、有用で一般の家庭装置のためのインタフェースを提供する。一シナリオで、電子レンジは料理アイテムパッケージ上のバーコード情報をスキャンでき、与えられた電子レンジシステムタイプに対して料理の調理時間を知るために製造業者データベースをアクセスできる。デバイス-デバイス命令及び制御を使用する装置の統合は、台所または居間で電話を受けた時自動で食器洗い器を止めテレビをミュート(Muting)させることのようなサービスを提供するために多くの制御シナリオを提供する。

'convenience'、これは例えばカーテン、窓、ブラインドまたは洗濯制御器に対するインタフェースのように便宜サービスを提供するために装置に対するインタフェースを提供する。

#### 【0099】

前記の説明で、'server\_av'はA/Vサービスを提供するA/V装置に対して制御インタフェースのための構造の一部分であって、'controls-gen'、'source'、'sink'性能にまた分けられる。

'controls-gen'は、デバイス製造者属性及びデバイス存在をテストするピング(Ping)テストのような一般のユーティリティインターフェーシングのためのインタフェースを提供する。さらに、ソフトウェア及びハードウェア確認及びバージョン情報のような製造特性が含まれる。前記インタフェースを提供するデバイスはどんな制御動作ももたらさずに前記ソフトウェアに対する名称または確認を提供するデータを返す。またデイクロックの時間をセットするインタフェースが含まれる。

'sink'は、メディアストリームサービスデバイスのためのインタフェースを提供する。その構造はVCRのような特別のデバイス名称というよりは提供されたサービス(例えば、ビデオストリーム録画及び再生)に基づいた構成である。例えば、チューナとDVDプレーヤーは二つともネットワークに対してビデオプログラムフォーマットを有するビデオプログラムストリームソースであって、開始及び停止のように制御される。特定のデバイス制御での差異点は構造化定義の下部階層によりアドレスされる。

'source'は'sink'インタフェースと類似のインタフェースを提供する。

### 【0100】

前述した内容を参照すれば、'service\_id'または'application\_interface\_id'は名称、アドレスまたはウェブアドレス、または一つ以上のデバイスのURL位置を含む。XCEデータベース104はインタフェースに対して合意された全体を構成するために、通常DHCP(Dynamic Host Configuration Protocol)ソフトウェアエージェントは各デバイスにアドレス及びデフォルト名称を割り当て、前記アドレス及びデフォルト名称はサービスまたはデバイスのインタフェースに追加される。それから、ソフトウェアエージェント110は'部分的なXCEネットワーク(Network partial XCE)'を生成するために、ホームネットワークに局部的に接続された全てのデバイスからサブセットまたは'部分的なXCEデバイス(Device partial XCE)'を含むデバイスインタフェース72を集める。付加的な関連外部インタフェースが外部制御のための構造に追加される。例えば、'service\_id'は、ネットワークに接続されたデバイスのデバイスインタフェースによってソフトウェアエージェントから加入を含む受信された構造またはネットワークインタフェースライブラリ106内の名称/アドレスでありうる。その後、使用者はデータベースでサービスを検索でき、前記名称/アドレスを使用してライブラリの特定のデータ部分を含むアプリケーションをアクセスできる。従って、ネットワークは前記名称/アドレス情報により識別された多重の同じサービスを含むことができる。

'media'、これは例えば、チューナからの伝送ストリーム、PC DRAMからRAM、DCまたはDVDのためのディスク及びテープを含むメディアタイプのためのインタフェースを提供する。前記メディアは命名され確認され、制御器デバイスはネットワーク上で提供される一般のメディアを確認するXCEデータベースを検索できる。DVDディスクのような新たなメディアがネットワーク上で提供される時、ディスク上でプログラム資料を確認するデバイスインタフェース72の一部が適合に変換される。従って、全体デバイスインタフェース72が伝送される必要はなく、ただ適した部分がXCEデータベースに伝送される。アテンション信号の受信でライブラリソフトウェアエージェント110は新たな更新をフェッチでき、適し

た場所にそれを位置させられる。ディスクメディアの付加は、前記ネットワークにサービスを追加したりまたは前記ネットワークに他の装置を接続することと類似である。

‘rate’、これは例えば、6Mbits/Secまたは19.2Mbits/Secのようにデバイスインタフェースのためのデータストリームレートに対する値を提供する。

‘protocol’、これは前記データストリームのために使われたプロトコルを確認する。例えば、61883/1394またはUI/IPのように、一つ以上のプロトコルが提供されれば、望ましいプロトコルが選択される。

‘stream\_format’、これはデジタルストリームオーディオ及びビデオ分割のためのパケットフォーマット及び/または圧縮標準を提供する。もし一つ以上のフォーマットが提供されれば、望ましいフォーマットがインタフェースメッセージを通じて選択される。制御器アプリケーション82は互換的なフォーマットがあるかどうかを決定できるフォーマットを調べられる。

‘controls\_av’、これはA/Vメディア装置のためのメイン制御インタフェースを提供する。

‘Flow\_control’、これは特定のデバイスのための方法であってPLAY、STOP、GO TO、RECORDのようなデータストリーム制御を提供する。その方法は、例えばPCソフトウェアを除いては埋め込まれた(Embedded)装置に対して変わらない。tkdrl制御は遅延された動作のために時間構成要素を含むことができる。

‘Tuning’、これはチューニング制御のためのインタフェースを提供する。制御器デバイス14は、制御されたデバイス14のインタフェースに前述された電子プログラムガイド(EPG: Electronic Program Guide)データ構造の返信を求められる。

‘UIcontrol’、これはディスプレイに対する輝度及びコントラスト、オーディオに対するボリューム及びベースのような修正を制御するために制御されたアプリケーション84に制御インタフェースを提供する。

‘Timer\_record’、これは遅延された時間記録を具現するために制御器アプリケーション82に対してセットアップデータのためのインタフェースを提供する。直接的なチャンネル同調情報及びフロー制御(time\_aparams)情報が使われたり

する。

### 【0101】

前記の説明はクライアントデバイス12に同一に適用できる。CE領域のための他のシンタックスXCE定義またはデータベースが使われたりする。代案的なシンタックスXCEデータベースは、例えば家庭自動化、装置及び自動車を含む全てのサービス説明を含む。サービス対象が制御のために適応性及びパラメータを提供する場合、制御方法は希望の通りに前記対象を制御するために使われる。2進数または16進数データ列を含むAV/C及びCAL命令語で命令例が下に示される。

```
consumer(document_file, doc)
    +---document_file<server_home.dtd, server_auto.dtd>
    +---doc(avec_commands, cal_commands, services_home, server_auto,)
server_auto,
server_samsung_web_site, server_auto_ford_explorer_98,,)
    +---avc_commands<...command_string...>
    +---cal_commands<...command_string...>
    +---services_home(client, av, lighting, comms, hvac, utility,
security, appliance, convenience,,)
        +---xml_utility(download_DTD_files,,)
        +---client(acknowledge, attention, error, post_schedule,
soundstop_schedule, stop_all,,)
            +---sound(alarm, ring, buzz,,)
            +---server_av(source, sink)
                +---source(service_id, media, rate, protocol,
stream_format, controls_gen, controls_av,,)
                    +---sink(service_id, media, rate, protocol,
stream_format, controls,,)
                        +---service_id(url,,)
                        +---media(tpt_stream, ram, disk, tape,,)
                        +---disk(name, number,,)
```

```

+---rate<value>
+---protocol(61883/1394,UDP/IP/Ethernet,,)
+---61883/1394(isoch_ch_no)
+---stream_format(video,audio,,)
+---video(dv,mpeg2tpt,dsstpt,mpeg2pes,
mpeg1080i-tpt,)
+---audio(mpeg3,ac-3,midi,,)
+---control_gen(ping,process_info,setup,,)
+---controls_av(flow_control,tune,timer_record,
ui_control,,)
+---process_info(s/w_id,h/w_id)
+---h/w_id(ser_no,manuf,model,class,,)
+---s/w_id(ser_no,exe_name,version,,)
+---setup(clock,,)
+---clock(hours,minutes,seconds)
+---time_record(tune,flow_control)
+---flow_control(play,stop,goto,record,,)
+---play(time_params)
+---record(time_params)
+---tune(send_epg,channel,,)
+---channel(number,id,
time_params,,)
+---time_params(now,start,
duration,end,,)
+---ui_control(display,acoustic)
+---display(brightness,contrast,
color/tint,horiz_size,vert_size,,)
+---acoustic(volume,bass,treble,
balance,fade,)

```

```

+----lighting(screen, light, send_epg)
+----sensors(living_room, sky,,)
+----lights(rooms_up, rooms_down, yard,,)
    +---rooms_up(bed1, bed2, bed3, bed4,,)
    +----rooms_down(family, kitchen, living, dining, soho,
garage,,)

    +----yard(front, back)
        +---bed1(lamp, dimmer,,)
        +----dimmer<value>
+----comms(netman, intercom, telco,)
    +---netman(send_device_list, send_configuration,
send_snmp_mib,,)

    +----intercom()
    +----telco()

+----hvac(controls_gen, controls_hvac,,)
    +---controls_hvac(a/c, heat, temp, humidity,)
        +----temp(low, high, hysteresis,,)
+----utility(meters, energy_mgmt,,)
    +---meters(water, gas, electric,,)
        +---water<value>, gas<value>,
electric<value>

    +----security(sensors, send_epg, alarm,,)
        +---sensors(peripheral, motion,,)
            +----peripheral(rooms_up, rooms_down,,)
            +----motion(rooms_up, yard,,)

    +----appliances(microwave, range, over, fridge, freezer, coffee,
toaster, washer, dryer, water-heater,,)
        +---microwave(send_epg, controls,,)
        +----fridge(temp,,)

```

```

+----water_heater(temp)

+----convenience(window,curtain_open,door/gate,pool/spa,
bath,fountain,lift,,)

+----curtain_open<value>

+----server_auto(message,mileage,maintenance,,)

+----mileage<data>

+----maintenance<data>

```

### 【0102】

他の観点で、本発明はネットワークでデバイス-デバイス命令及び制御のために存在する命令語具現の使用を提供する。デバイスは、内部対象及び存在する伝送メカニズムによって、実行時に2進数列を生成するAPIを有する。このような場合、ネットワークで一つのデバイス14からさらに他のデバイス14にXML遠隔手順呼び出し(XMLRPC: XML Remote Procedure Calls)を提供するために、存在するアプリケーションインタフェース具現はXMLサービスAPIに対する呼び出しに取り替えられる。従って、本来の具現はXMLサービスAPIのためのラップ(Wrapper)と同等である。図18は、またXCE/XMLサービスAPIでラップに取り替えられたインタフェース具現として、点線内のCALまたはAV/Cのような他の命令語を使用して生成されたアプリケーションを示す。CAL命令語からXMLRPCフォーマットを変える例が下に示されている。

---

existing implementation:

```

void DeviceCALCommand(int command){
    ×
    ×
    /*
    create CAL formatted byte string to represent this
    object/method and output to the wire
    */
    CreateCALFormattedByteString(command);/* different for

```

```

        every protocol*/

        SendCALByteString(); /*different for every protocol

*/
}

```

---

wrapping the XML Service API call:

```

void DeviceCALCommand(int command){
{
    /*
        replace CAL implementation with calls to the XML
ServiceAPI
    */
    CreateXMLMessage(command); /* always the same */
    sendXMLMessage();          /* always the same */
}

```

### 【0103】

さらに他の観点で、図23を参照すれば、本発明はネットワーク内の他のデバイス間のデバイス間通信のための標準命令プロトコル及び制御言語翻訳を提供する。情報を共有する他のデバイスのために、前記情報は要請デバイスが解釈できるフォーマットであるべきである。そして、一つのデバイス120が他のデバイス22を制御するために、前記二つのデバイスはお互いの命令を解釈するために共通の言語を使用すべきである。本発明はデータのための共通識別フォーマット及び命令プロトコルを提供する。

### 【0104】

一実施形態で、共通表示またはデータ及び命令プロトコルパッケージする方法が提供され、それにより受信デバイス122は伝送されたデータの本来のフォーマットを決定できる。もし受信デバイスが本来のフォーマットを解釈できるならば、その時にデータを直接受信できる。そうでなければ、受信デバイス122は翻訳デバイス124または要請デバイス122が解釈できる望ましいフォーマット

トにデータを変形するアプリケーションを求められる。前記翻訳デバイス124またはアプリケーションは元のデータの本来のフォーマットを決定し、前記データを前記求められるフォーマットで翻訳し、要請デバイス122に翻訳したデータを送信する。

#### 【0105】

次に、前記要請デバイス122は、まるで前記データが元々送信デバイス120により要請デバイスの本来の言語フォーマットで提供されたようにデータ进行处理する。前記要請デバイス122は、また要請デバイスの本来のフォーマットで送信デバイス120に返信を送ったり翻訳デバイス124または送信デバイス120の本来のフォーマットで翻訳するアプリケーションを通じてプロキシにより応答を伝送できる。前記翻訳方法は、命令プロトコル、データファイル及びオーディオ/ビデオストリームを含む情報のために使われる。

#### 【0106】

前述された共通フォーマットを使用しないデバイスのために、本発明は固い(Non-compliant)デバイスへ/からの命令プロトコルを含むデータの翻訳を提供する。例えば、固いデバイス120が柔軟なデバイス122にデータを送信する時、前記柔軟なデバイス122は前記データの本来のフォーマットの決定に基づいてデータを翻訳できる。例えば、前記柔軟なデバイス122はデータ内部の特定のビットパターンのためのデータを調査できる。柔軟なデバイスは既知の固いデバイスにデータを伝送する時、前記柔軟なデバイスは固いデバイスの本来のフォーマットの決定に基づいて伝送する前に前記データを翻訳できる。

#### 【0107】

例示的な具現はIP及びHTTPプロトコルを支援するホームネットワークのことでありうる。前記ホームネットワークは望ましい機能のためのアプリケーション及び多様なタイプのサービスを得るためにインターネットに接続されたりする。従って、共通フォーマット方法は、インターネット及びホームネットワーク上の動作に対するインターネットプロトコル及び手順と互換性を有するように作られる。

#### 【0108】

共通データフォーマットを提供する一例は、ホームネットワークを通じて伝送のためのデータパッケージを生成するXMLを使用している。前記データは命令プロトコル、オーディオまたはビデオストリーム、グラフィックまたはアプリケーションを含む。前記データはXML形態でデータの本来のフォーマット及びパッケージの項目を確認する標準ヘッダでラッピングされる。前記ヘッダは唯一にデータタイプとXMLコードのデータ部分を確認するようにし、それによって前記データは必要に応じて翻訳され、受信後に適当なアプリケーションに提供される。

#### 【0109】

ウェブ標準下で、前記確認作業はファイル伝送のタイプ及び項目を確認するためにファイル名拡張を使用するブラウザにより行われる。次に、ブラウザはそのファイルを処理する適当なプラグ-インのモジュールを移す。ホームネットワークでXMLは、前述された共通確認方法でIP上の全てのホームネットワーク伝送を提供するデータ伝送を確認するために使われる。

#### 【0110】

一方、ホームネットワークを通した全てのデータ伝送の項目を唯一に確認するソフトウェア階層がホームネットワークプロトコルスタックで提供される。前記ソフトウェア層はXMLの代りに使われる。本発明の共通フォーマット及び確認原理は、XMLを使用する実施形態または確認方法としての前記ソフトウェア階層に同一に適用される。

#### 【0111】

図23でデータパッケージ伝送を受信すれば、受信デバイス122はその中のデータのフォーマットを決定するデータパッケージのXML確認ヘッダを調べる。もし前記データがデバイス122により認識できるフォーマットであれば、XML確認ヘッダ情報は捨てられ、前記デバイスはデータを直接処理する。そうでなければ、前記デバイス122は受信されたXMLパッケージをXML翻訳要請パッケージに変換し、翻訳サーバデバイス124に要請パッケージ及び前記データを送信する。

#### 【0112】

前記翻訳サーバデバイス124は前記データを翻訳し、前記翻訳したデータを

XML翻訳応答パッケージに変換する。次に、前記翻訳サーバ124は要請デバイス122に応答パッケージを回送する。翻訳エラーの場合に、前記翻訳サーバ124は要請デバイス122に翻訳応答エラー状態を提供できる。翻訳されたデータを受信すれば、前記要請デバイス122は翻訳されたデータを応答パッケージで処理する。

### 【0113】

XMLデータパッケージまたはパケットの例は次の通りである：

```
<IDENTITY type=format=AV/c>...packet data...</IDENTITY>
```

翻訳要請パッケージまたはパケットの例は次の通りである：

```
<TRANSLATION REQUEST type=Command format=CAL>
```

```
<IDENTITY type=Command format=AV/C>...packet data
```

```
</IDENTITY>
```

```
</TRANSLATION REQUEST>
```

翻訳要請パッケージまたはパケット例は次の通りである：

```
<TRANSLATION RESPONSE type=Command format=CAL>...packet data
```

```
</TRANSLATION RESPONSE>
```

伝送応答エラー状態パッケージまたはパケットの例は次の通りである：

```
<TRANSLATION RESPONSE type=Command format=CAL>...packet data...
```

```
<ERROR condition=Unrecognized command>Translation could not be
performed</ERROR>
```

```
</TRANSLATION RESPONSE>
```

### 【0114】

さらに、図24でテーブル3は、パッケージまたはパケットタイプ及びフォーマットの部分的なリストを含む。

### 【0115】

翻訳サービスを提供するために、翻訳サーバ124はDHCPサーバの方法と類似な方法でネットワーク構成の間にネットワークで確認される。前記翻訳サーバ124は、ネットワークが構成された後一定期間ネットワーク内の全てのデバイスにそのIPアドレスを放送する。翻訳サービスと互換できる全てのデバイス120

、122は、ネットワークをブーティングする間にネットワークを通じて放送する翻訳サーバ124のIPアドレスを貯蔵する。

【0116】

一方、要請デバイス122はホームネットワークを通じて翻訳要請を放送できる。ネットワークで翻訳要請を受信する全ての翻訳サーバ124は要請デバイス122に翻訳応答を送信することによって翻訳要請に応答できる。次に、要請デバイス122は応答翻訳サーバ中で一つの翻訳サーバ124を選択する。一例として、要請デバイス122は翻訳要請に応答する最初の翻訳サーバ124を選択する。また他の例で翻訳サーバ124は、翻訳要請を満たす翻訳サーバ124の選択のためにかれら自体間でそして/または要請デバイス122と交渉できる。

【0117】

本発明のさらに他の実施形態で、多重の翻訳サーバ124が全ての翻訳要請を行うために使われる。例えば、単一の翻訳サーバ124が全ての要請を翻訳する性能を有しない場合もありうる。この場合、各翻訳サーバのアドレス及び各翻訳サーバが提供できる翻訳サービスのタイプを確認する必要がある。各デバイス120、122は、全ての翻訳サーバIPアドレスリストと各翻訳サーバ124が提供する翻訳サービスタイプに相応するリスト、そして選択的に関連された翻訳アプリケーションを貯蔵できる。

【0118】

効率のために、もし送信デバイス120が送信デバイス120フォーマットと違う本来フォーマットを使用することと知られた受信デバイス122にデータを伝送することを希望するならば、前記送信デバイス120は翻訳サーバ124を通じたプロキシにより受信デバイス122にデータを伝送できる。前記送信デバイス120は、翻訳サーバ124に伝送要請命令と類似に翻訳サーバ124に命令を伝送し、翻訳されたデータに対する目的地として受信デバイス122のアドレスを含む。

【0119】

受信デバイス122がデータストリームの翻訳を要求する場合には、送信デバイス120は直接翻訳サーバ124にデータストリームをルーティングでき、前

記翻訳サーバ124は前述されたように受信デバイス122に翻訳されたデータを順次に伝送する。他の一方では、前記送信デバイス120は前記受信デバイス122にデータストリームを伝送でき、次に受信デバイス122は翻訳及び前記受信デバイス122に翻訳されたデータを回送するために翻訳サーバ124にデータストリームをルーティングする。

#### 【0120】

説明中で、制御メカニズムは分配され、協調的なハイパーメディア情報システムのためのアプリケーションレベルプロトコルを提供するHTTP1.1に基づく。HTTPは多くのタスクのための幅広い使用で一般的で無状態(stateless)で対象向きのプロトコルである。HTTPの特徴は、伝送されたデータと独立的にシステムが構成されるようにするデータ表現の分類及び交渉である。望ましくは、ホームネットワーク上でデバイス及びアプリケーションにより使われたネットワークプロトコルはIP(Internet Protocol)である。しかし他のプロトコルも使われる。

#### 【0121】

本発明は望ましい実施の形態で詳細に説明されたが、他の変形も可能である。従って、特許請求の範囲は本発明の望ましい実施の形態により限定されてはいけない。

#### 【図面の簡単な説明】

【図1】 本発明の第1観点に係るネットワークの具現のためのブロック図である。

【図2】 デバイス制御及び通信概要の第1実施形態に係る図1のブロック図である。

【図3】 多数のクライアントとサーバデバイスを含む本発明に係るホームネットワークシステムに対する第1実施形態を示すブロック図である。

【図4】 図3のクライアントデバイス及びサーバデバイス具現の第1実施形態を示すブロック図である。

【図5】 クライアントデバイス具現の第1実施形態を示す図面である。

【図6】 サーバデバイス具現の第1実施形態を示す図面である。

【図7】 相互間に通信及び制御を可能にするネットワークサーバデバイス

の第2実施形態を示すブロック図である。

【図8】 ネットワークでソースサーバデバイス、シンクサーバデバイス及びクライアントデバイスを含むオーディオ/ビデオモデルの構造に対する第1実施形態を示すブロック図である。

【図9】 オーディオ/ビデオモデルの他の実施形態を示す図面である。

【図10】 ネットワークデバイスのための性能データテーブルの実施形態を示す図面である。

【図11】 図11は、ネットワークデバイスのための特性データテーブルの実施形態を示す図面である。

【図12】 ネットワークデバイス間で命令メッセージを発生させるビルディングブロックに対する第1実施形態の構成を示す図面である。

【図13】 命令メッセージを発生させる図12に係るビルディングブロックの他の実施形態の構成を示す図面である。

【図14】 ネットワーククライアント及びサーバデバイス間で相互作用の第3実施形態を示す図面である。

【図15】 ネットワークデバイスインタフェースのAPI拡張を定義する第1実施形態のブロック図である。

【図16】 他のサーバデバイスのインタフェース説明文書をアクセスするサーバデバイスアプリケーションのための第1実施形態の構造を示す図面である。

【図17】 制御器サーバデバイスと制御されたサーバデバイスとの間の内部デバイス制御構造の他の実施形態を示す図面である。

【図18】 ネットワークデバイス間のAPIレベルで通信スタックでウェブ標準共通ミドルウェア階層を提供するXMLプロトコルの具現を示す図面である。

【図19】 サーバデバイス間命令制御構造のまた他の具現を示す図面である。

【図20】 デバイスインタフェースライブラリとホームデバイスのための電化製品データベースとの関係を示す図面である。

【図21】 デバイスインタフェース定義に対する階層的形態の具現を示す

図面である。

【図22】 図21のデバイスインタフェース定義で階層の第1実施形態を示す図面である。

【図23】 伝送器及び受信器デバイス間で共通伝送及び解釈手順を示す図面である。

【図24】 本発明の第1観点に係る翻訳サービスを提供するためのパケットタイプとフォーマットの部分的なリストテーブルの第1実施形態を示す図面である。

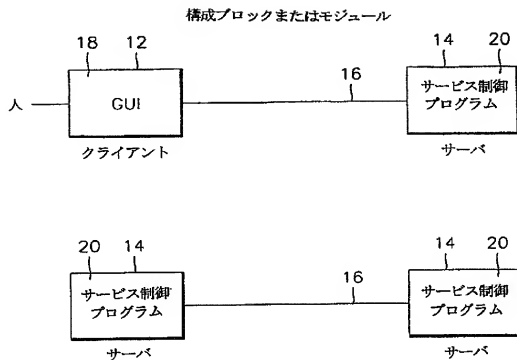
【符号の説明】

- 10 ネットワーク
- 12 クライアントデバイス
- 14 サーバデバイス
- 16 通信リンク
- 18 GUIディスプレイ
- 20 サーバ制御プログラム
- 22 グラフィック制御対象(GCO) 使用者インタフェース説明書
- 24 クライアントデバイスレンダラ
- 26 制御状態データ
- 28 クロック
- 30 命令語インタフェース
- 32 データストリームソースハードウェア
- 34 データストリームシンクハードウェア
- 36 セッション管理者
- 40 サービスの機能的規格ブロック
- 42 メッセージ要素が構成されるブロック
- 44 産業標準化フォーマットブロック
- 46 メッセージ圧縮ブロック
- 48 メッセージ文字列構成ブロック
- 50 ソフトウェアクライアント制御プログラム

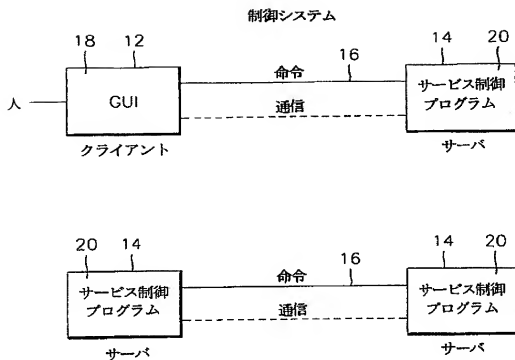
- 52, 54, 58, 60 ブロック
- 56, 62 ルックアップテーブル
- 64 ライブラリデバイス
- 66 通信スタック
- 68 XML層OUT
- 70 XML層IN
- 72 XMLインタフェース
- 74 XMLパーサ
- 76 XMLRPCコーデック
- 78 インタフェースフェッチャ
- 79 ホームネットワーク対象要請ブローカ
- 80 ホームネットワークインタフェースライブラリ
- 82 制御器アプリケーションプログラムコード
- 84 アプリケーションプログラムコード
- 86 ホームネットワークデバイスウェブサーバ
- 88 固有ルックアップテーブル
- 90 ハンドラ
- 92 ハードウェアサービス
- 94 固有インタフェース
- 96 第3デバイス
- 98 ミドルウェア階層
- 100 データベース
- 102 CEデータベース
- 104 XCEデータベース
- 106 ホームネットワークインタフェースライブラリ
- 108 製造会社手段
- 112, 114, 116, 118 階層
- 120 送信デバイス
- 122 要請デバイス

## 124 翻訳デバイス

【図1】

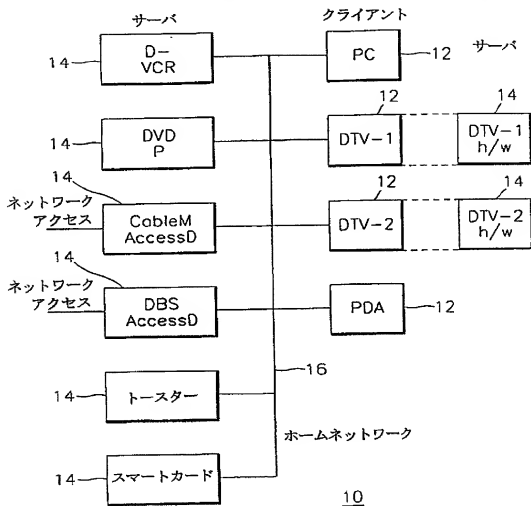


【図2】



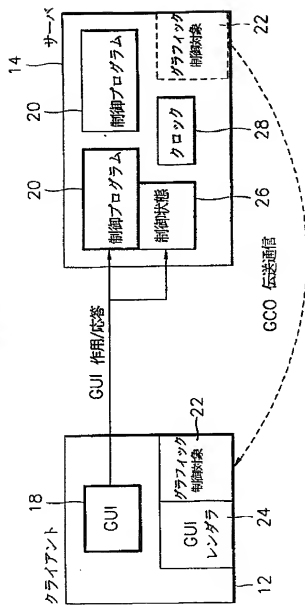
【図3】

複数のクライアント 複数のサーバ ホームネットワーク制御モデル



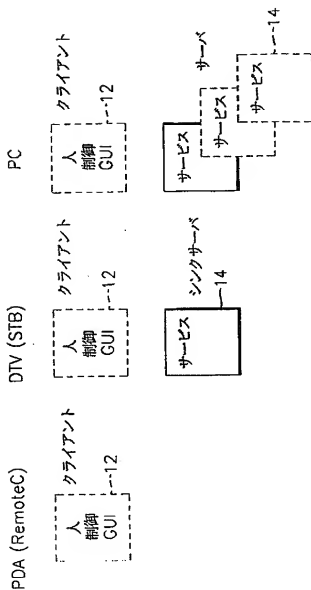
【図 4】

## モジュール構成 GUI 制御

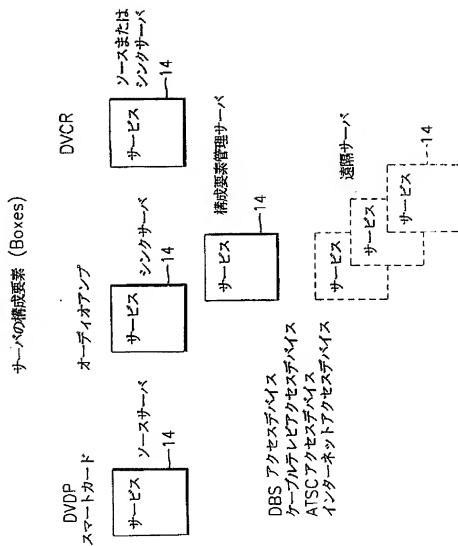


【図5】

クライアントの構成要素

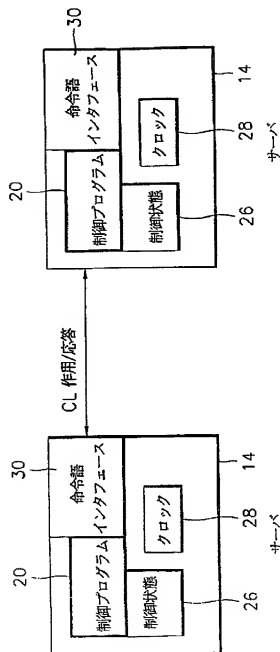


【図6】

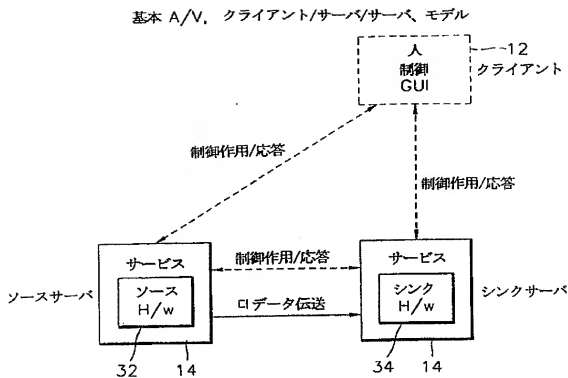


【図7】

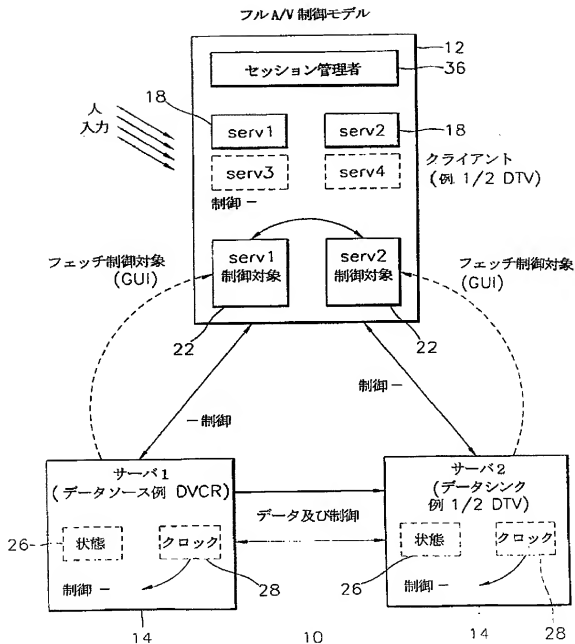
サーバ/サーバCLのためのモジュール構成



【図8】



【図9】



【図 10】

テーブル1：性能テーブル

Service	Sink/Source
Service A	Sink
Service B	Source
Service C	Source
Service D	Sink

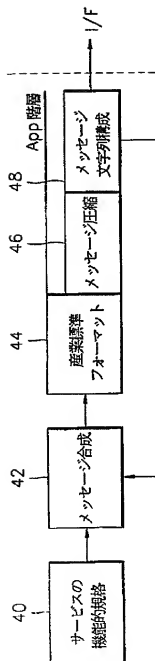
【図11】

テーブル2：特性テーブル

名称	長さ	値
DeviceManufacturer	20 chars	Device manufacturer's name
ManufacturerURL	60 chars	device manufacturer's home page URL
manufacturerIcon	20 chars	Name of Device manufacturer's icon
DeviceName	30 chars	Device name
DeviceModel	20 chars	Device model number
DeviceType	20 chars	device category
DeviceLocation	30 chars	device group or location
DefaultSource	15 chars	Data type, Default source device (IP address)
DefaultSink	15 chars	Data type, Default sink device (IP address)
DeviceIcon	20 chars	Name of device icon

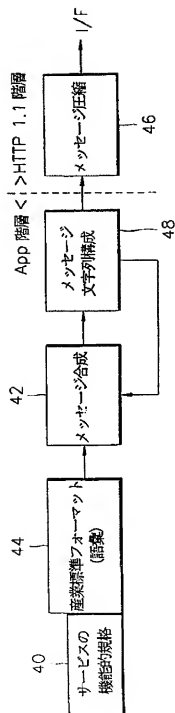
【図 12】

CAN 及び AV/C により使用されたメッセージ構成

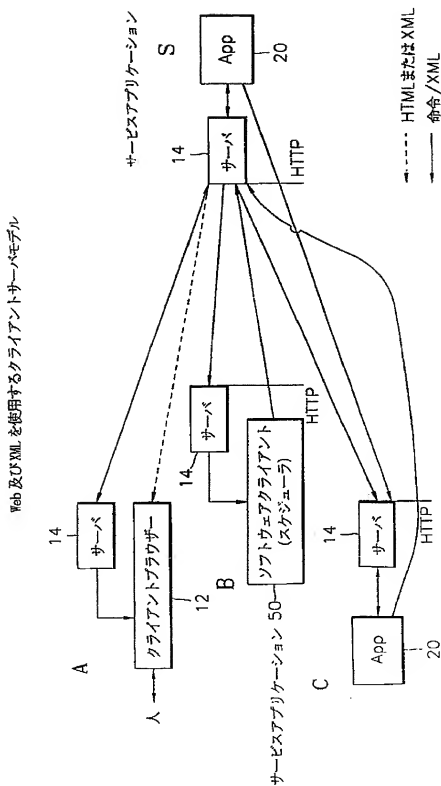


【図13】

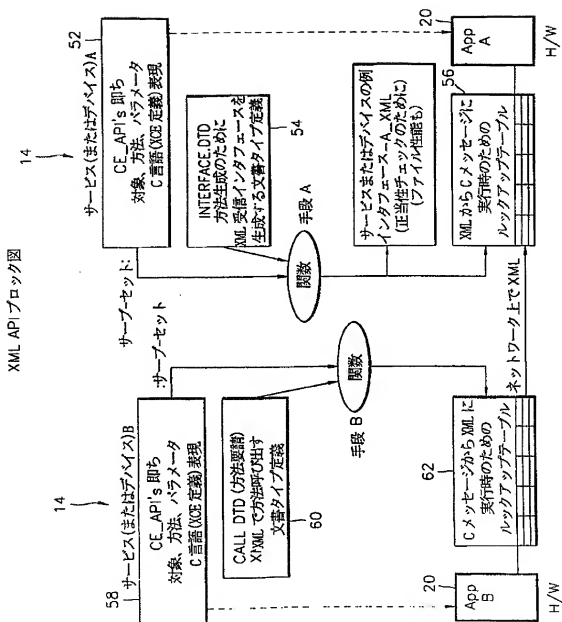
新たなXCE/XMLのメッセージ構成



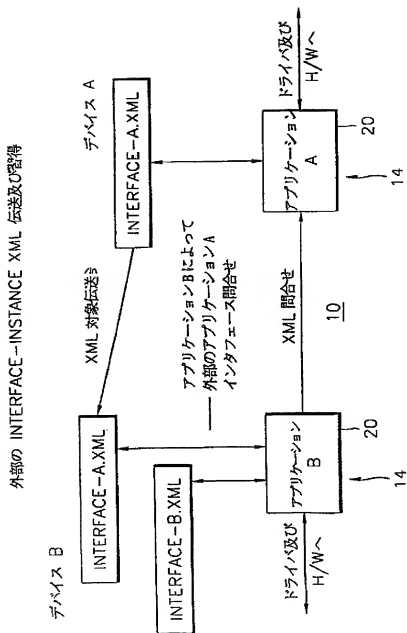
【図 14】



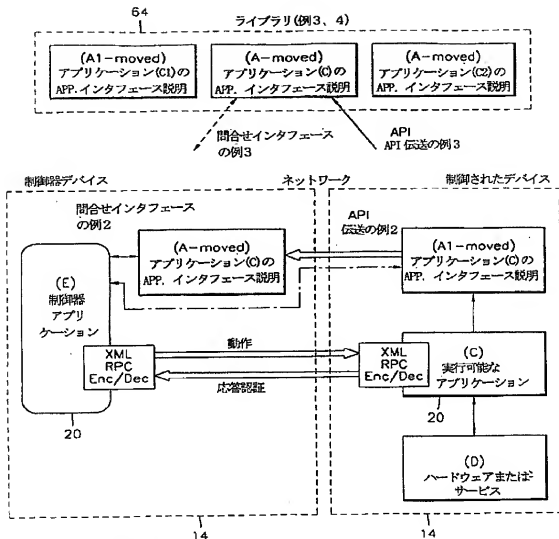
【図15】



【図16】



【図 17】



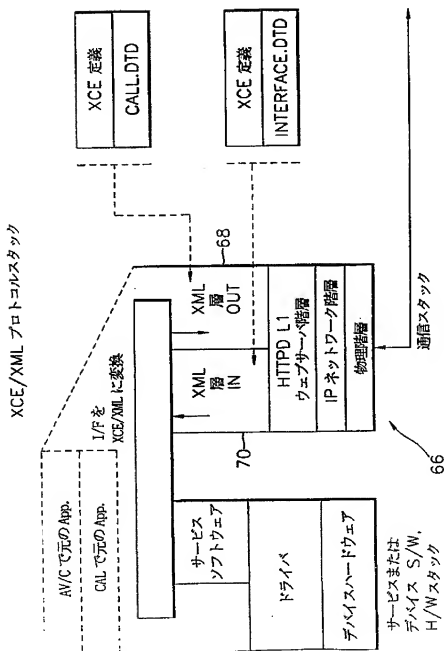
例1. (E)により遠隔で問合せするためのXMLアプリケーションCインタフェース(A):

例2. (E)により局所的に問合せするためのXMLアプリケーションCインタフェース(A)を制御側デバイスに移動

例3. (E)により遠隔で問合せするためのXMLアプリケーションCインタフェース(A)を第3デバイス全てのインタフェースのための集合場所(ライブラリ)に移動  
ライブラリは直接制御動作及び応答が有用に關したアプリケーションの住所(URL)を有すべきである

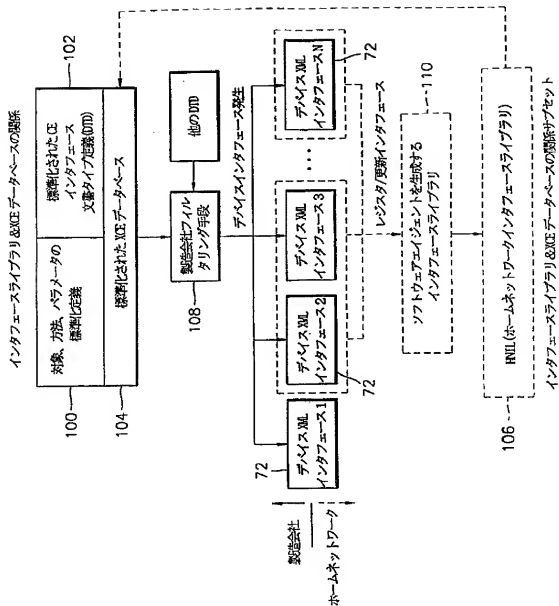
例4. 例3と同じであるが、"間接的な制御動作(及び応答)"がライブラリデバイスで命令される。この場合にライブラリは"送達者と共に使われる"

【図18】

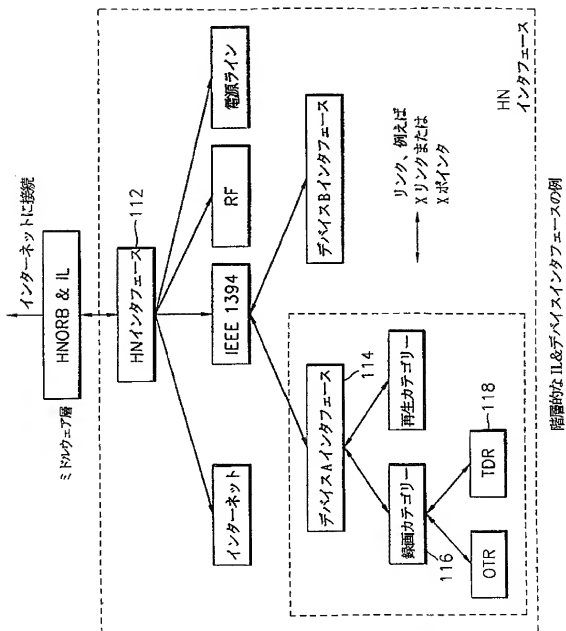




【図20】



【図21】



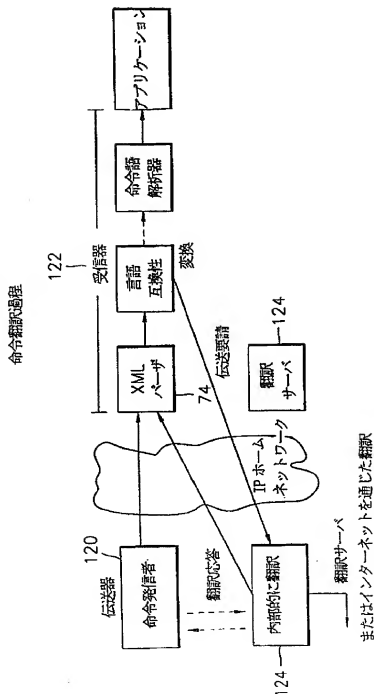
階層的な IL &amp; デバイスインタフェースの例

【図22】

階層的なインタフェース構造、4階層

階層	インタフェース階層	インタフェース例
112 — 1	各HJのためのXMLインタフェース、 現在有用なデバイスリスト	My HN (VCR, TV ...)
114 — 2	各デバイスのための一般のインタフェース、 機能カテゴリ—リスト	一般のVCRインタフェース(録画、再生...)
116 — 3	デバイスに対する各機能カテゴリ—のための XMLインタフェース規格	VCR、録画、カテゴリ—
118 — 4	機能カテゴリ—で各機能のための XMLインタフェース規格	VCR、録画、tdd 機能

【図 23】



【図24】

テーブル3：フォーマットとタイプ例

タイプ	フォーマット
命令	CAL, AV/C, X-10
映像	jpeg, gif, bitmap, tiff
言語	English, French, etc
ビデオ_ストリーム	mpeg2
ビデオ_クリップ	avi, quicktime, mpeg
テキスト	html, plain
オーディオ	wav, aiff
アプリケーション	maword, pdf, postscript, gzip

## 【国際調査報告】

## INTERNATIONAL SEARCH REPORT

International application No.  
PCT/KR 99/00221

## A. CLASSIFICATION OF SUBJECT MATTER

IPC<sup>7</sup>: H 04 N 7/00

According to International Patent Classification (IPC) or to both national classification and IPC

## B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

IPC<sup>7</sup>: H 04 N; H 04 L; G 06 F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

EPODOC

## C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	US 5546584 A (LUNDIN et al.) 13 August 1996 (13.08.96), abstract; fig. 1-4	1,9,12
A	WO 96/21189 A1 (COMPUSEVE) 11 July 1996 (11.07.96), claim 1; fig. 1A.	9
A	EP 0814403 A1 (H.P) 29 December 1997 (29.12.97), abstract; fig. 1.	1
A	EP 0784271 A2 (NEC) 16 July 1997 (16.07.97), claim 3; fig. 1.	1

☐ Further documents are listed in the continuation of Box C.☒ See patent family annex.

\* Special categories of cited documents:

- "A" documents defining the general state of the art which is not considered to be of particular relevance
- "B" earlier application or patent but published on or after the international filing date
- "C" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
- "D" document relating to an oral disclosure, use, exhibition or other means
- "E" documents published prior to the international filing date but later than the priority date claimed

- "F" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principles or theory underlying the invention
- "X" documents of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
- "Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
- "Z" document member of the same patent family

Date of the actual completion of the international search

19 April 2000 (19.04.00)

Date of mailing of the international search report

03 May 2000 (03.05.00)

Name and mailing address of the ISA/AT  
Austrian Patent Office  
Kohlmarkt 8-10; A-1014 Vienna  
Postcode No. 1/53424/200

Authorized officer

Mihatsek

Telephone No. 1/53424/329

Form PCT/ISA/210 (second sheet) (July 1998)

**INTERNATIONAL SEARCH REPORT**  
 Information on patent family members

 International application No.  
**PCT/KR 99/00221**

Patent document cited in search report	Publication date	Patent family members	Publication date
US A 5544884	13-08-1996	AB A1 45869/93	31-01-1994
		AG B2 606185	05-02-1998
		BR A 9306654	08-17-1998
		CV A 10887053	12-01-1994
		EP A3 648334	19-04-1995
		FI A 946194	30-12-1994
		IT A0 946118	30-12-1994
		MX A1 9303456	28-02-1994
		NO A0 945054	27-12-1994
		NO A 945054	06-02-1993
		NO B1 9401829	20-01-1994
		NO B1 44252/96	24-07-1996
WO A1 9621189	11-07-1996	US B 5680549	21-10-1997
		US B 6092862	18-12-1999
		US A1 10105/97	17-07-1997
		AU B2 710044	09-09-1999
		CA A0 2194320	23-07-1997
		JP B2 9190431	22-07-1997
EP A1 814403	29-12-1997	US B 5680549	21-10-1997
EP A2 784211	16-07-1997	AU B2 710044	09-09-1999
EP A3 784211	21-01-1998	JP B2 9190431	22-07-1997
		US B 5680549	21-10-1997

## フロントページの続き

(81)指定国 EP(AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OA(BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG), AP(GH, GM, KE, LS, MW, SD, SL, SZ, UG, ZW), EA(AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, CA, CH, CN, CU, CZ, DE, DK, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MD, MG, MK, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, UA, UG, UZ, VN, YU, ZW  
Fターム(参考) 5B089 GA11 GA23 GB03 HA18 HB02  
JA35 JB02 LB01

【公報種別】特許法第17条の2の規定による補正の掲載

【部門区分】第6部門第3区分

【発行日】平成19年12月6日(2007.12.6)

【公表番号】特表2002-514797(P2002-514797A)

【公表日】平成14年5月21日(2002.5.21)

【出願番号】特願2000-547722(P2000-547722)

【国際特許分類】

G 0 6 F 13/00 (2006.01)

【F I】

G 0 6 F 13/00 3 5 7 A

【誤訳訂正書】

【提出日】平成19年10月18日(2007.10.18)

【誤訳訂正1】

【訂正対象書類名】明細書

【訂正対象項目名】特許請求の範囲

【訂正方法】変更

【訂正の内容】

【特許請求の範囲】

【請求項1】 ホームネットワーク上でサービスを行うための方法において、

(a) 第1ホームデバイスを前記ホームネットワークに接続する段階と、

(b) 第2ホームデバイスを前記ホームネットワークに接続する段階と、

(c) 複数のアプリケーションインタフェース説明データ対象を含むデータベースを提供する段階とを**含み**、各アプリケーションインタフェース説明データ対象は前記ネットワークに接続された一つまたはそれ以上の他のホームデバイスによるホームデバイスの命令及び制御のための情報を含み、

(d) 前記第1ホームデバイスが前記データベースで前記第2ホームデバイスのためのアプリケーションインタフェース説明データ対象をアクセスする段階と、

(e) 前記第1ホームデバイスが前記ネットワークを通じて前記第2ホームデバイスのために前記アプリケーションインタフェース説明データ対象を使用して前記第2ホームデバイスに命令及び制御データを伝送する段階とをさらに含み、それにより前記第1及び第2ホームデバイスが直接的に前記サービスを行うことを特徴とするホームネットワーク上でサービスを行うための方法。

【請求項2】 前記構造化されたフォーマットはXMLフォーマットを含むことを特徴とする請求項1に記載の方法。

【請求項3】 前記データベースは前記ネットワークに接続されたデータベースデバイスに含まれることを特徴とする請求項1に記載の方法。

【請求項4】 (i) 前記第1ホームデバイスは前記第1アプリケーションインタフェース説明データ対象を貯蔵し、

(ii) 前記第2ホームデバイスは前記第2アプリケーションインタフェース説明データ対象を貯蔵し、

(iii) 前記(c)段階は、前記データベースデバイスに前記第1及び第2ホームデバイスのための前記アプリケーションインタフェース説明データ対象を伝送するために前記第1及び第2ホームデバイスを問い合わせることを含む段階により前記データベースを形成する初期化段階を含むことを特徴とする請求項3に記載の方法。

【請求項5】 前記(d)段階は、ネットワークを通じて前記データベースから前記第2ホームデバイスに前記第1ホームデバイスのための第1アプリケーションインタフェース説明データ対象を提供する段階を含むことを特徴とする請求項1に記載の方法。

【請求項6】 前記(e)段階は、ネットワークを通じて前記データベースから前記第

1 ホームデバイスに前記第2 ホームデバイスのための第2 アプリケーションインタフェース説明データ対象を提供する段階を含むことを特徴とする請求項1に記載の方法。

【請求項7】 三つまたはそれ以上のホームデバイスを前記ネットワークに接続する段階として、少なくとも一つのホームデバイスは前記ネットワークを通じて複数のホームデバイスに命令及び制御データを伝送するための複数のホームデバイスのアプリケーションインタフェース説明データ対象を問い合わせるためにデータベースをアクセスする段階をさらに含むことを特徴とする請求項1に記載の方法。

【請求項8】 各アプリケーションインタフェース説明データ対象は構造化されたフォーマットでデータを含むことを特徴とする請求項1に記載の方法。

【請求項9】 サービスを提供するためのネットワークシステムにおいて、

(a) 相互通信するデバイスにより使われる通信媒体を提供する物理層と、

(b) 第1 ホームデバイスと、

(c) 第2 ホームデバイスと、

(d) 多数のアプリケーションインタフェース説明データ対象を含むデータベースとを含む、

それぞれのアプリケーションインタフェース説明データ対象は、前記ネットワークに接続された一つまたはそれ以上の他のデバイスによるホームデバイスの命令及び制御のための構造化されたフォーマットでの情報を含む。

前記第2 ホームデバイスは、前記データベースで前記第1 ホームデバイスのための第1 アプリケーションインタフェース説明データ対象をアクセスし、前記第1 アプリケーションインタフェース説明データ対象を使用して前記第2 ホームデバイスから前記第1 ホームデバイスに制御及び命令データを伝送するためのアプリケーション制御手段を含む、

前記第1 ホームデバイスは、前記データベースで前記第2 ホームデバイスのための第2 アプリケーションインタフェース説明データ対象をアクセスし、前記第2 アプリケーションインタフェース説明データ対象を使用して前記第1 ホームデバイスから前記第2 ホームデバイスに制御及び命令データを伝送するためのアプリケーション制御手段を含む、

それにより前記第1 及び第2 ホームデバイスは自律的に前記サービスを行うことを特徴とするサービスを提供するためのネットワークシステム。

【請求項10】 前記構造化されたフォーマットはXMLフォーマットを含むことを特徴とする請求項9に記載のネットワークシステム。

【請求項11】 前記データベースはデータベースデバイスに貯蔵されることを特徴とする請求項9に記載のネットワークシステム。

【請求項12】 (i) 前記第1 ホームデバイスは第1 アプリケーションインタフェース説明データ対象を貯蔵し、

(ii) 前記第2 ホームデバイスは第2 アプリケーションインタフェース説明データ対象を貯蔵し、

(iii) 前記データベースデバイスは、データベースデバイスに各々前記第1 及び第2 アプリケーション説明対象を伝送するために前記第1 及び第2 ホームデバイスを問い合わせることによって前記データベースを形成することを特徴とする請求項11に記載のネットワークシステム。

【請求項13】 前記第2 ホームデバイスのアプリケーション制御手段は、前記データベースから第1 ホームデバイスのための第1 アプリケーションインタフェース説明データ対象を得ることを特徴とする請求項9に記載のネットワークシステム。

【請求項14】 前記第1 ホームデバイスのアプリケーション制御手段は、前記データベースから第2 ホームデバイスのための第2 アプリケーションインタフェース説明データ対象を得ることを特徴とする請求項9に記載のネットワークシステム。

【請求項15】 少なくとも一つのホームデバイスは、ネットワークを通じて複数のホームデバイスに命令及び制御データを伝送するための複数のホームデバイスのアプリケーションインタフェース説明データ対象を問い合わせるためにデータベースをアクセスする三つまたはそれ以上のホームデバイスをさらに含むことを特徴とする請求項9に記載の

ネットワークシステム。

【請求項 16】 それぞれのアプリケーションインタフェース説明データ対象は構造化されたフォーマットでデータを含むことを特徴とする請求項 9 に記載のネットワークシステム。

【請求項 17】 前記構造化されたフォーマットは XML フォーマットを含むことを特徴とする請求項 9 に記載のネットワークシステム。

【請求項 18】 ネットワーク上でサービスを行うための方法において、

(a) デバイスが前記ネットワークに接続されたときにアプリケーションインタフェース説明データを取得するために前記デバイスに問い合わせる段階を含み、前記アプリケーションインタフェース説明データは前記ネットワークに接続された他のデバイスによる前記デバイスの命令及び制御のための情報を含み、

(b) 前記取得したアプリケーションインタフェース説明データをデータベースに貯蔵する段階と、

(c) 第 1 デバイスのアプリケーションインタフェース説明データを前記データベースから第 2 デバイスに提供する段階とをさらに含み、前記第 2 デバイスは命令及び制御のために前記第 1 デバイスのアプリケーションインタフェース説明データを使用して前記第 1 デバイスと相互作用することを特徴とするネットワーク上でサービスを行うための方法。

【誤訳訂正 2】

【訂正対象書類名】明細書

【訂正対象項目名】0002

【訂正方法】変更

【訂正の内容】

【0002】

【従来の技術】

一般的に、ネットワークは通信リンクと前記通信リンクと接続された通信能力を有する多様なデバイスを含む。前記デバイスは、コンピュータ、周辺装置、ルータ、貯蔵装置、そしてプロセッサと通信インタフェースを具備した製品を含む。ネットワークの一実施形態として多様なデバイスが相互接続された家庭用ホームネットワークを挙げられる。普通の家庭はパソコン及び家庭内で典型的に見つけられるホームデバイスを含むいろいろなデバイスを含むことができる。用語としての“デバイス”は、典型的に論理デバイスまたは機能性及びデータを交換する能力を備える他の装置を含み、全てのホームデバイスだけでなく汎用コンピュータも含むことができる。ホームデバイスは保安システム、映画館装備、TV、VCR、ステレオ装備、デジタル衛星サービス(DSS: Digital Satellite Services)とも知られる衛星放送サービス(DBSS: Direct Broadcast Satellite Services)、散水システム、照明システム、電子レンジ、食器洗い器、オープン/ストープ、洗濯機/乾燥器及び自動車内の処理システムのような電子デバイスを含む。

【誤訳訂正 3】

【訂正対象書類名】明細書

【訂正対象項目名】0024

【訂正方法】変更

【訂正の内容】

【0024】

クライアントデバイス 12 とサーバデバイス 14 は、例えば DTV のように一つのユニットで物理的に一緒に結びつけることができる。その場合に、前記サーバデバイス 14 はサーバハードウェアを制御するための制御プログラム 20 を含み、前記クライアントデバイス 12 は少なくともサーバハードウェアの制御及び命令のためにサーバ制御プログラム 20 に制御用インタフェースを提供する。図 5 はクライアントデバイス 12 の例を示す図面であって、(1) GUI ディスプレイのための PDA (RemoteC)、(2) GUI をディスプレイし、オーディオ及び/またはビデオプログラムストリーム目的地サーバで構成された受信側サーバを含む DTV (STB)、(3) GUI をディスプレイし、多重のサービスを提供するために少

なくとも一つのサーバデバイスを含むPCを含むことができる。DTVまたはPCクライアントデバイス内のハードウェアあるいは実行できるものは他のクライアントデバイスによっても制御される。図6はサーバデバイス14の例を示す図面であって、(1) 送信側のサーバデバイスとしてのDVDP SmartCard、(2) 受信側のサーバデバイスとしてのオーディオ増幅器、(3) 送信側または受信側のサーバデバイスとしてのDVCR、(4) 遠隔サーバデバイスを管理するための管理サーバを含む。前記管理サーバは例えばDSB-STB、ケーブルTV-STB、またはATSC-STBを含むことができる。このようなデバイスはSTB内部作業のローカル制御または管理のための管理サーバを含む。さらに、外部ネットワークを通じてアクセスされた外部サーバは、例えばVideo-on-Demand、Enhanced-TV及びインターネット商取引のようなサービスのためにローカルクライアントデバイスにより使われる場合がある。

【誤訳訂正4】

【訂正対象書類名】明細書

【訂正対象項目名】0028

【訂正方法】変更

【訂正の内容】

【0028】

図8は、ネットワークで送信側のサーバデバイス14、受信側のサーバデバイス14、そしてクライアントデバイス12を含むオーディオ/ビデオ(A/V)モデルの例を示す。前記送信側のサーバデバイス14は送信側のサーバデバイス14のデータストリーム送信側ハードウェア32を制御するための制御プログラム20を含み、前記受信側のサーバデバイス14は受信側のサーバデバイス14のデータストリーム受信側ハードウェア34を制御するための制御プログラム20を含む。例示的な動作で、使用者はデータストリーム送信側ハードウェア32を稼働させるために送信側のサーバデバイス14を制御し、データストリーム受信側ハードウェア34を稼働させるために受信側のサーバデバイス14を制御するクライアントデバイス12を使用する。データストリーム送信側ハードウェア32からデータストリーム受信側ハードウェア34へのデータ伝送を初期化することにおいて、使用者はこれ以上前記クライアントデバイス12を操作する必要がない。他的一方では、使用者は将来の初期データ伝送をプログラムでき、これ以上クライアントデバイス12を操作する必要がない。以後で送信側のサーバデバイス14のデータストリーム送信側ハードウェア32と受信側のサーバデバイス14のデータストリーム受信側ハードウェア34は使用者によりプログラムされた時間に自動でデータ伝送を初期化する。

【誤訳訂正5】

【訂正対象書類名】明細書

【訂正対象項目名】0029

【訂正方法】変更

【訂正の内容】

【0029】

例えば、前記データストリーム送信側ハードウェア32は、衛星放送(DBS: Direct Broadcast Satellite)のようなチューナーアクセスデバイスを含むことができる。DBSはケーブルテレビに代える多重チャンネルであって、衛星から直接的に小さな衛星アンテナ(直径3フィート18インチ)にケーブルのようなテレビプログラミングを提供する。DBSと共に、いくつかの標準アナログテレビ信号は単一衛星トランスポングでデジタル的に圧縮され、空中の指定された場所に位置した衛星アンテナも200つ以上のチャンネルが受信することを許容する。前記データストリーム受信側ハードウェア34は、再生時に圧縮されたデジタルビデオ信号をデコーディングできる命令及びデジタルビデオカセットレコーダ(DVCR)を含むことができる。使用者はDVCRのための“時間-遅延録画”イベントデータとチューナーアクセスデバイスのための“時間-遅延選択プログラム”イベントデータを含む命令及び制御データを提供する。時間遅延後に前記チューナーアクセスデバイスは希望のプログラムを選択し、使用者から追加的な制御操作なしでもプログラムデータを受信及び録画するDVCRにソースプログラムデータを提供する。

【誤訳訂正 6】

【訂正対象書類名】明細書

【訂正対象項目名】0030

【訂正方法】変更

【訂正の内容】

【0030】

図9は、少なくともネットワークで送信側のサーバデバイス14 SERVER1、受信側のサーバデバイス14 SERVER2とクライアントデバイス12を含むさらに他のA/Vモデルを示す図面である。前記クライアントデバイス12は、サーバデバイス14 SERVER1及びSERVER2と他のサーバデバイス14、例えばSERVER3及びSERVER4（図示せず）を選択し制御するために、使用者のための選択情報をディスプレイする使用者インタフェースのあるセッションマネージャ36を含む。選択情報はサーバデバイス14 SERVER1、SERVER2、SERVER3及びSERVER4を各々選択する使用者のためにセッションマネージャ36でServ1、Serv2、Serv3及びServ4に割当てられたアイコンシンボルを含む。前記送信側のサーバデバイス14 SERVER1はDVCRを含むことができ、前記受信側のサーバデバイス14 SERVER2は1/2 DTVを含むことができる。

【誤訳訂正 7】

【訂正対象書類名】明細書

【訂正対象項目名】0031

【訂正方法】変更

【訂正の内容】

【0031】

動作の一例として、サーバデバイス14 SERVER1とSERVER2の選択において、前記クライアントデバイス12は各サーバデバイス14のGC022をクライアントデバイスに伝送し各サーバデバイス14 SERVER1及びSERVER2に対応するGUI18をディスプレイする。使用者は、サービスのために対応するサーバデバイス14に制御と命令を提供する各サーバデバイス14のGUI18と相互作用できる。各サーバデバイス14は単独または他のサーバデバイス14と組合してサービスを提供できる。さらに、前記セッションマネージャ36は、サービスを行うために対応するサーバデバイス14が必要とするものであって、前記クライアントデバイス12で前記サーバデバイス14のGUI18間に制御状態データ26を伝送する。使用者制御と命令情報に基づいて、二つ以上のサーバデバイス14は使用者が求めるサービスを提供するためにそれら間で命令及び制御情報を通信できる。

【誤訳訂正 8】

【訂正対象書類名】明細書

【訂正対象項目名】0032

【訂正方法】変更

【訂正の内容】

【0032】

前記セッションマネージャ36は、ネットワーク10で多様なサーバデバイス14により提供される有用なホームネットワークサービスをアクセスしディスプレイする機能を有するソフトウェアエージェントを含むことができる。前記ソフトウェアエージェントは付加的にネットワーク10で多様なサーバデバイス14の性能を整合させ、互換性能を有する前記サーバデバイス14だけのための選択情報をディスプレイする。さらに、セッションマネージャ36は、使用者がサーバデバイス14に意味のある命令及び制御情報を提供するように一つのサーバデバイス14のGUI18で作られた選択と、他のサーバデバイス18のGUI18で作られた選択を整合させられる。

【誤訳訂正 9】

【訂正対象書類名】明細書

【訂正対象項目名】0033

【訂正方法】変更

## 【訂正の内容】

## 【0033】

動作のさらに他の例で、セッションマネージャ36はネットワークを検索し、ネットワークに接続されたサーバデバイス14を見つけるソフトウェアエージェントを実行する。前記ソフトウェアエージェントは、またサーバデバイス14の性能を決定するために各サーバデバイス14に貯蔵された性能データをアクセスし、使用者にその性能に対する情報を提供する。それから前記セッションマネージャ36は、図9に示したようにサーバデバイスSERVER1、SERVER2、SERVER3及びSERVER4のための選択アイコンServ1、Serv2、Serv3及びServ4をディスプレイする。

## 【誤訳訂正10】

【訂正対象書類名】明細書

【訂正対象項目名】0034

【訂正方法】変更

## 【訂正の内容】

## 【0034】

前記セッションマネージャ36は、使用者が4個の全てのアイコン中で選択できるように初期に全ての選択アイコンServ1、Serv2、Serv3及びServ4をイネーブルさせる。使用者がServ1選択アイコン上でクリックしてサーバデバイスSERVER1を選択した後に、前記セッションマネージャ36はサーバデバイスSERVER3及びSERVER4がサーバデバイスSERVER1と性能上互換できないと決定する。従って、前記セッションマネージャ36はサーバデバイスSERVER3及びSERVER4のために選択アイコンServ3及びServ4を各々ディスエーブルさせる。それから使用者はサーバデバイスSERVER2を命令及び制御するためにアイコンServ2上でクリックできる。

## 【誤訳訂正11】

【訂正対象書類名】明細書

【訂正対象項目名】0035

【訂正方法】変更

## 【訂正の内容】

## 【0035】

使用者が選択されたサーバデバイス14のGUI18と相互作用することのように、使用者により各GUI18に入力された制御及び命令情報は使用者により後続するサーバデバイス選択に影響を及ぼす付加的な性能情報を提供する。例えば、もしVCRサーバデバイス14が選択されれば、他のサーバデバイス14のための選択アイコンをイネーブルまたはディスエーブルさせることにおいて、セッションマネージャ36による後続の操作が再生または録画する使用者の決定により影響される。

## 【誤訳訂正12】

【訂正対象書類名】明細書

【訂正対象項目名】0037

【訂正方法】変更

## 【訂正の内容】

## 【0037】

各サーバデバイス14はネットワーク上でいろいろな他のサービスのために送信または受信するので、図10に示したように各サーバデバイス14は性能データテーブル(性能テーブル1)を貯蔵する。テーブル1の第1列はサーバデバイス14のサービス性能を確認し、第2列は第1列に対応するサービスのために前記サーバデバイス14が送信側か受信側かを確認する。新たなサービスはさらに古いデバイスと互換性を維持する間に性能データテーブル1を使用して実行される。例えば、もしさらに古いサービスと互換される新たなサービスが開発されるならば、新たなサービス及び古いサービス両方とも実行デバイスは古いサービスを使用する古いデバイスと互換性を維持するにつれて、前記新たなサービスを行うデバイスのための性能データテーブル1に書き込まれることができる。

【誤訳訂正 13】

【訂正対象書類名】明細書

【訂正対象項目名】0038

【訂正方法】変更

【訂正の内容】

【0038】

第1実施形態で、デバイスマネージャは送信側及び受信側のサービスデバイスのマッチングまたは比較を行う。例えば、前記デバイスマネージャは互換性マッチングと共に多様なサブデバイス14及び指定されたサブデバイス14の性能及び特性を比較するソフトウェアエージェントの役割も行える。例えば、サービスがネットワークにわたって第1サブデバイス14から第2サブデバイス14までメディアストリームの場合、前記デバイスマネージャは使用者が第1サブデバイス14の性能と互換される第2サブデバイス14を分別力あるように選択できるように第1及び第2サブデバイス14の性能を比較する。次はサブデバイス14の実施形態のためのサービス性能リストの例を示す。

```
Stream_format_video_dv
Stream_format_video_mpeg2tpt
Stream_format_video_dsstpt
Stream_format_video_mpeg2pes
Stream_format_video_mpeg21090i-tpt
```

【誤訳訂正 14】

【訂正対象書類名】明細書

【訂正対象項目名】0039

【訂正方法】変更

【訂正の内容】

【0039】

図11に示したように、各サブデバイス14はデバイスに関する特性を含む特性データテーブル(特性テーブル2)をさらに貯蔵する。名称と値はテーブル2で各特性を定義する。文字長さがテーブル2に示されているが、必要ではない。前記特性データはネットワーク10上で情報処理相互運用を容易にしデバイス情報を貯蔵する他のサブデバイス14に有用である。例えば、後述されるようにデバイスページはデバイス名称を貯蔵する特性テーブル2を使用する。他のフィールドは必要な時特性データテーブル2に追加されることができる。

【誤訳訂正 15】

【訂正対象書類名】明細書

【訂正対象項目名】0041

【訂正方法】変更

【訂正の内容】

【0041】

特性テーブル2で前記デバイス位置特性フィールドは各サブデバイス14のための位置またはグループを貯蔵するために使われる。デバイスタイプ特性フィールドはVCR、DVD、DTV、カムコーダ、PC、保安システムのような特定のサブデバイス14のためのデバイスタイプを書き込む。前記デバイスタイプ特性フィールドは、もしデバイス自体がデバイスアイコンを供給しないならば、デバイスページでデバイスを表現するデフォルトデバイスアイコンを選択するために使われる。特性テーブル2はデフォルトソース及びデフォルトシンク特性フィールドのための多重の書き込みを含むことができる。そのようなそれぞれの書き込みは、サブデバイス14により処理される各データタイプのための他のデフォルトの送信側または受信側のサブデバイス14を表現する。

【誤訳訂正 16】

【訂正対象書類名】明細書

【訂正対象項目名】0045

【訂正方法】削除  
 【訂正の内容】  
 【誤訳訂正17】  
 【訂正対象書類名】明細書  
 【訂正対象項目名】0046  
 【訂正方法】削除  
 【訂正の内容】  
 【誤訳訂正18】  
 【訂正対象書類名】明細書  
 【訂正対象項目名】0047  
 【訂正方法】削除  
 【訂正の内容】  
 【誤訳訂正19】  
 【訂正対象書類名】明細書  
 【訂正対象項目名】0049  
 【訂正方法】変更  
 【訂正の内容】  
 【0049】

第2実施形態" B" で、使用者はソフトウェアクライアント制御プログラム50により置き換えられる。前記ソフトウェアクライアント制御プログラム50はXML基盤のサービスアプリケーション" S" への命令ポストティングを生成し、返信されるXML命令ポストティングを受信する。そして、第3実施形態" C" で、前記ソフトウェアクライアント制御プログラム50は、命令と応答が二つのサービスアプリケーション20間で交換されるサーバデバイス制御プログラム20のようなアプリケーションにより置き換えられる。この点において、実施形態" B" はナル(Null)サービスを有する実施形態" C" の特別な場合である。

【誤訳訂正20】  
 【訂正対象書類名】明細書  
 【訂正対象項目名】0052  
 【訂正方法】変更  
 【訂正の内容】  
 【0052】

サービスAのためのAPI拡張を参照すれば、第1最上位ブロック52は、CEデバイスを説明する英単語を使用してCE対象及び方法の総合的な定義またはデータベースを提供する。前記総合的な定義またはデータベースはC、XMLまたは対象及びそれぞれ各自の方法を表現できる他のフォーマットでも存在できる。XMLを使用する総合的な定義またはデータベースはXCE定義と呼ばれる。第2ブロック54は、インタフェースデータタイプ定義INTERFACE.DTDでデザインされた全てのサーバデバイス14のためにXML形式でAPIを表現するフォーマットを提供する。

【誤訳訂正21】  
 【訂正対象書類名】明細書  
 【訂正対象項目名】0064  
 【訂正方法】変更  
 【訂正の内容】  
 【0064】

図17を参照すれば、制御器サーバデバイス14及び制御されたサーバデバイス14間のデバイスサーバデバイスまたは内部デバイスを制御する他の例が示される。前記制御器サーバデバイス14は制御器アプリケーションを含み、制御されたサーバデバイス14は実行可能なアプリケーションCを含む。制御されたサーバデバイス14はINTERFACE-A.XML、アプリケーションCのアプリケーションインタフェース説明をさらに含む。アプリケーション

ンEは、制御されたサーバデバイス14の性能及び制御されたサーバデバイス14のAPIインタフェース方法を問い合わせるために制御されたサーバデバイス14内のアプリケーションインタフェース説明Aをアクセスする。次に、アプリケーションEは、制御されたサーバデバイス14のハードウェアまたはサービスDを制御するためにXML遠隔手順呼び出しを使用してアプリケーションCを命令し制御する。スケジューラデバイスはVCRの時間一遅延一録画制御器のように定められた日になれば駆動する制御器サーバデバイス14の一場合でありうる。

【誤訳訂正22】

【訂正対象書類名】明細書

【訂正対象項目名】0065

【訂正方法】変更

【訂正の内容】

【0065】

第1例で、前記アプリケーションEは、ネットワークを通じた遠隔照会によりアプリケーションインタフェース説明Aをアクセスする。第2例で、前記アプリケーションEは、制御されたサーバデバイス14から制御器サーバデバイス14にアプリケーションインタフェース説明Aのコピーを伝送することによってアプリケーションインタフェース説明Aをアクセスする。次に、アプリケーションEは局部的にインタフェース説明Aに問い合わせる。第3例で、アプリケーションインタフェース説明Aはインタフェース説明のためにライブラ空間を提供するライブラデバイス64に伝送され、アプリケーションEはライブラにインタフェース説明Aを遠隔で問い合わせる。前記ライブラデバイス64は直接的な制御動作及び応答を使用できる関連したアプリケーションのアドレス(URL)を貯蔵する。

【誤訳訂正23】

【訂正対象書類名】明細書

【訂正対象項目名】0066

【訂正方法】変更

【訂正の内容】

【0066】

図18を参照すれば、XMLプロトコルはネットワークで多様なサーバデバイス14のアプリケーション20間のAPIレベルで通信スタック66においてウェブ標準共通ミドルウェア階層を提供する。各サーバデバイス14で通信スタックの最上部アプリケーションはネットワークを通じて通信メッセージを送受信し、デバイスのためのデバイスハードウェアまたはサービスソフトウェアを局部的に制御するデバイススタックでソフトウェア階層と通信する。

【誤訳訂正24】

【訂正対象書類名】明細書

【訂正対象項目名】0069

【訂正方法】変更

【訂正の内容】

【0069】

それぞれのサーバデバイスA及びBはネットワークを通じて他のサーバデバイスを制御し、ネットワークを通じて他のサーバデバイスにより制御されるためのハードウェアとソフトウェアを含む。図19で、ホームネットワークデバイスAは制御器デバイスまたはモジュールであり、ホームネットワークデバイスBは制御されたデバイスまたはモジュールである。それぞれのデバイスA及びBは、インタフェース文書INTERFACE.XML及び文書タイプ定義INTERFACE.DTDで構成された局所的なデバイスXMLインタフェース72を含む。INTERFACE.XML文書は対応するサーバデバイス14により支援される対象、方法及びパラメータの説明を含む。前記INTERFACE.DTD文書は、前述されたようにデバイスのXMLインタフェースで具体的な正当性チェックのために使われたりする。

【誤訳訂正 251】

【訂正対象書類名】明細書

【訂正対象項目名】0071

【訂正方法】変更

【訂正の内容】

【0071】

プログラムコードで構成されたインタフェースフェッチャ(Fetcher)は、さらに他のデバイスまたはホームネットワークインタフェースライブラリ80から直接的にさらに他のデバイスのデバイスインタフェースをフェッチするためにそれぞれのデバイスA及びBにより使われる。サーバデバイス14が制御器デバイスであれば、制御器サーバデバイス14内の制御器アプリケーションプログラムコード82は、XMLパーサ74、インタフェースフェッチャ78及びXMLRPCコーデック76のような制御器サーバデバイス14内のソフトウェア及びハードウェアを統制することによって、ネットワークを通じて他のサーバデバイス14の命令及び制御に影響を与える。サーバデバイス14が制御されたデバイスであれば、制御されたサーバデバイス14内の制御されたアプリケーションプログラムコード84は他のサーバデバイス14により制御されるサーバデバイス14のためにサーバデバイス14内のソフトウェア及びハードウェアを統制する。それぞれのデバイスA及びB内のホームネットワークデバイスウェブサーバ86はネットワーク上のデバイス間の通信を管理する。各デバイスA及びB内の固有ルックアップテーブル88に対するXMLはX、MLRPCメッセージ(例えば方法名称、構成要素名称及びタイプ)をデバイスの固有インタフェース(例えば固有方法名称、構成要素及びタイプ)に変換する制御されたアプリケーション84により使われる。前記テーブル88は、XMLメッセージ内方法及びパラメータの名称とデバイスの固有インタフェースが同じ場合には使われない。

【誤訳訂正 261】

【訂正対象書類名】明細書

【訂正対象項目名】0073

【訂正方法】変更

【訂正の内容】

【0073】

さらに、ホームネットワーク対象要請ブローカ(HNORB: Home Network Object Request Broker)79及びインタフェースライブラリ(IL: Interface Library)のようなネットワーク対象要請ブローカはホームネットワーク10のためのミドルウェア階層98を提供する。図19に示したように、前記ミドルウェア階層98は第3デバイス96または分離された制御ハブに位置できる。前記HNORB79は、ネットワーク10に接続された他のサーバデバイス14の存在を見つける一つのサーバデバイス14により使われるためのソフトウェアエージェントを含む。前記HNORBソフトウェアエージェントはデバイス名称を名称の階層的なツリー構造で組織し、デバイスインタフェースを前記検索できるインタフェースライブラリで組織し、そしてインタフェース情報を要請するデバイスにデバイスインタフェースを提供する。

【誤訳訂正 271】

【訂正対象書類名】明細書

【訂正対象項目名】0075

【訂正方法】変更

【訂正の内容】

【0075】

前記インタフェースライブラリ80を使用するために、少なくとも一つのHNORB&ILは局所的なホームネットワーク10上で動作中であるべきである。一つ以上のHNORB&ILがまた利用される。例えば、ケーブルモデム、いくつかのDTV、そして中央家庭ハブは全てそれら自体のHNORB&ILソフトウェアエージェントを有することができる。HNORB&ILの位置を定めるためにサーバデバイス14は局所的なホームネットワークを通じて放送メッセージを

伝送する。サーバデバイス 14 で応答した第 1 HNORB&IL はサーバデバイス 14 により使われる。一応 HNORB&IL が位置すれば、サーバデバイス 14 及び HNORB&IL は登録、インタフェース要請及びフェッチのための地点間 TCP (Transmission Control Protocol) または UDP (User Datagram Protocol) とデバイスルックアップサービスを確認できる。もし UDP プロトコルを利用できなければ、TCP プロトコルが IEEE 1394 のような高帯域幅接続のために使われる。HTTP-基盤の XMLRPC は HNORB&IL 通信のためのデバイスのために使われる。例えば、サーバデバイス 14 は一つ以上の構成要素としてデバイスインタフェースをパスするために HNORB の "レジスタ" 方法を遠隔で呼び出すことができ、または XMLRPC 呼び出しは XMLRPC 応答または戻る値として IL から部分的または全体的なデバイスインタフェースを再生できる。

【誤訳訂正 28】

【訂正対象書類名】明細書

【訂正対象項目名】0076

【訂正方法】変更

【訂正の内容】

【0076】

前述したように、一つ以上の HNORB&IL は同時に局所的なホームネットワーク 10 で動作でき、ここで各 HNORB&IL は使用可能なデバイスのサブセットを認識でき、一つの HNORB&IL は見つけられないサーバデバイス 14 を位置させるために他の HNORB&IL と通信できる。一つの局所的なホームネットワーク 10 上の多重 HNORB&IL は、UDP 及び TCP のような放送メッセージを使用することによって自動で相互間に位置できる。このような場合に、多重 HNORB&IL は多数のインタフェースライブラリ 80 が分散インタフェースライブラリを構成する間分散対象要請プロコラを構成する。故障許容誤差を提供するために、もし一つの HNORB&IL が突然終了しなければならないならば、この HNORB&IL と共に登録された全てのデバイスは通知され、前記デバイスは他の有用な HNORB&IL と共に自動で登録される。

【誤訳訂正 29】

【訂正対象書類名】明細書

【訂正対象項目名】0082

【訂正方法】変更

【訂正の内容】

【0082】

望ましくは、全てのデバイスがネットワークを通じて RPC 呼び出しを説明しデコーディングできるように、標準 XMLRPC フォーマットが使われる。制御されたサーバデバイス 14 のデバイスインタフェースが制御器サーバデバイス 14 により問い合わせられ調査されるために、望ましくは能率を向上させるために十分なデバイスインタフェース情報を有する単純化された XMLRPC フォーマットが使われる。次の例は OTR (One Touch Record) 及び TDR (Time Delayed Record) 動作のために XMLRPC を呼び出す二つの可能なフォーマットを示す。

【誤訳訂正 30】

【訂正対象書類名】明細書

【訂正対象項目名】0085

【訂正方法】変更

【訂正の内容】

【0085】

図 20 を参照すれば、ホームデバイスであるサーバデバイス 14 のためのデバイスインタフェースは、標準化された用語を使用する産業標準の構造化されたデータベース 100 に基づく。新たなインタフェースのためのインタフェースデータ及び用語はデータベース 100 に追加される。CE 対象を説明するために英単語を使用する包括的な定義あるいはデータベース、方法そしてパラメータは CE データベース 102 に規定される。包括的な定義またはデータベースは対象及びそれらそれぞれの方法及び構成要素を示す、XML または他のフォーマットになりうる。標準化された XML 用語を使用する包括的な定義またはデータ

ベースはXCE定義またはデータベース104と呼ばれる。

【誤訳訂正31】

【訂正対象書類名】明細書

【訂正対象項目名】0087

【訂正方法】変更

【訂正の内容】

【0087】

図20で、埋め込まれた装置14のために'製造会社'情報としてデザインされた情報は製造時装置14に内蔵されたものであり、'ホームネットワーク'としてデザインされた情報はネットワークで装置の動作の実行時間特徴の一部である。Nデバイスのための1...NとしてデザインされたデバイスXMLインタフェース72は、標準化されたXCEデータベース104でデータの一部分である。ホームネットワークインタフェースライブラリ(HNIL: Home Network Interface Library)106はホームネットワークに接続された使用可能なサーバデバイス14のデバイスインタフェース集合を提供する。HNIL106は全体XCEデータベース104のサブセットである。

【誤訳訂正32】

【訂正対象書類名】明細書

【訂正対象項目名】0089

【訂正方法】変更

【訂正の内容】

【0089】

図20を参照すれば、前記XCEデータベース104は、またCEサーバデバイス14を表現するXMLを使用するために標準化された規定セットを提供するCEデバイスのための標準化されたXCEインタフェース文書タイプ定義(DTD: Document Type Definition)を提供する。前記DTDまたはそのサブセットは正当性チェックのために使われる。製造会社手段108としてデザインされたソフトウェアエージェントは、規定されたCEデバイスのための標準化されたXCE定義のサブセットをフィルタリングして使用するが、例えばINTERFACE.XML及びINTERFACE.DTDのようなCEデバイスのXMLデバイスインタフェース72を生成する標準化されたXCEインタフェースDTDを使用する。前記文書INTERFACE.XMLは標準化されたXCEインタフェースDTDに従って特殊なデバイスにより支援される対象、方法及びパラメータの説明を含む。前記文書INTERFACE.DTDは標準化されたXCEインタフェースDTDのサブセットであって、デバイスのXMLインタフェースのための正当性チェックのために使われる。他の文書タイプ定義もINTERFACE.XML文書を生成するために使われたりする。

【誤訳訂正33】

【訂正対象書類名】明細書

【訂正対象項目名】0090

【訂正方法】変更

【訂正の内容】

【0090】

前記XMLインタフェース文書及び前記DTD文書を含むCEデバイスのXMLインタフェース72は、ホームネットワークインタフェースライブラリ106のような一般的にアクセスできるライブラリに貯蔵される。ソフトウェアエージェント110はネットワーク上の全てのアクセスできるサーバデバイス14のデバイスインタフェース72を収集し、デバイス名称/アドレス情報と共に検索できる構造化されたインタフェースライブラリ106にそれらを配置する。前記インタフェースライブラリ106はXCEデータベース104のサブセットであって、インタフェースライブラリ106を作る過程は部分または全てのXCEデータベース104を再構成することと類似である。前記インタフェースライブラリ106は、ホームネットワーク内の全てのデバイスのデバイスインタフェースの集合または最も最近に使われたデバイスインタフェース72が貯蔵されるホームネットワークで全てのサーバデバイス14の有用性に依存するキャッシュ(Cache)として機能できる。サーバデバ

イス14が、DVD再生器でディスクを変えることのようなイベントに起因してそのデバイスインタフェース72を更新した場合には、デバイスインタフェース72の一部がイベントサービスに基づいて更新される。

【誤訳訂正34】

【訂正対象書類名】明細書

【訂正対象項目名】0091

【訂正方法】変更

【訂正の内容】

【0091】

図21を参照すれば、望ましくは各サーバデバイス14のデバイスインタフェース定義72は階層的な形態を有する。なぜならこれはホームデバイスにおいて、デバイスインタフェース定義72が長くなる可能性があるからである。典型的に、時間遅延された録画のための単一機能のような一つまたは少数の機能は同時にアクセスされ、従って前記デバイスインタフェース72の唯一に小さな部分が使われる。全体デバイスインタフェース72を作成することよりはデバイスインタフェース72の一部分だけを作成することがより効果的である。階層的なデバイスXMLインタフェースを使用することによって、制御器サーバデバイス14は、制御器サーバデバイス14またはHNRB及びILミドルウェア階層98からXMLデバイスインタフェースのための要求において望ましい機能カテゴリーまたは機能を特定することによって、制御されたサーバデバイス14の部分的なデバイスインタフェース72を求められる。後者の場合に、HNRB及びILミドルウェア階層98はデバイスインタフェース72の望ましい部分を回送する。

【誤訳訂正35】

【訂正対象書類名】明細書

【訂正対象項目名】0094

【訂正方法】変更

【訂正の内容】

【0094】

本発明に係る他のサーバデバイス14を制御するサーバデバイス14のための典型的な命令及び制御モデルで、第1サーバデバイス14は第2インタフェース階層114で第2サーバデバイス14のデバイスインタフェースを問い合わせるように試みる。機能カテゴリー(FC: Function Category)を選択した後、前記第1サーバデバイス14は、録画カテゴリーのような第2サーバデバイス14で特定の機能カテゴリーのインタフェース階層116を問い合わせる。さらに、前記第1サーバデバイス14は前記機能に対する呼び出しを作るためにOTRまたはTDRのように特定の機能をインタフェース階層118に問い合わせる。階層的またはツリー構造はより効果的なインタフェース機能を探すようにしてネットワーク帯域幅を縮める。インタフェースファイル構造及び層の例が下記の通りである。

第1層 112 - HN1.xml XML

第2層 114 - VCRI.xml

第3層 116 - VCRI\_RecordCategory.xml

第4層 118 - VCRI\_RecordCategory\_OTR.xml

【誤訳訂正36】

【訂正対象書類名】明細書

【訂正対象項目名】0097

【訂正方法】変更

【訂正の内容】

【0097】

前記階層的なデバイスインタフェース定義72は次のような分野を含む。

'document file'、これはXCEデータベース104またはXCEデータベース104のバージョン部分の正当性及び訂正の検証のためにXMLパーサにより使われる文書タイプの規定(DTD)ファイルの名称を提供する。前記XCE構造の相異なる部分のためのいろいろなDTD

ファイルが考えられ、ここで前記DTDはRPC CALLのための文書タイプ定義及び通信のためのINTERFACE.DTDとは違う。

'doc'、これは互換性、特性、通信及び制御インタフェースの領域の最上位レベル名称を提供する。

'Services\_home'、これは家庭自動化、電化製品、効用性のための領域を提供する。

'Server\_auto'、これは車庫内にある自動車に対して、一台またはそれ以上の自動車タイプに対する有用なメッセージインタフェースを示す。例えば、'server\_auto\_ford\_explorer\_98'は特定自動車に対するインタフェースである。これは自動車のマイレージ及び保守インタフェースをアクセスできるようにし、直接チェックする自動車製造会社または車庫が直接チェックまたは遠隔診断する遠隔アクセスで使われる。

'server\_samsung\_web\_site'、これは家庭外部の製造会社ウェブサイトと通信し、メッセージ、サービス、アシストなどのためのインタフェースを含む。

'AVC\_commands'及び'CAL\_commands'、これは例えばAV/C及びCAL言語を解釈できるレガシーデバイスを提供する。この構造部分は前記言語で命令を確認し、命令はXMLで付加されたり行われる。従って、前記項目はXCE(Web)対象ではなく、プロトコル変換器アプリケーションは元のCALまたはAV/Cアプリケーションソフトウェアへのインタフェースとして使われる。

【誤訳訂正37】

【訂正対象書類名】明細書

【訂正対象項目名】0100

【訂正方法】変更

【訂正の内容】

【0100】

前述した内容を参照すれば、'service\_id'または'application\_interface\_id'は名称、アドレスまたはウェブアドレス、または一つ以上のデバイスのURL位置を含む。XCEデータベース104はインタフェースに対して合意された全体を構成するために、通常DHCP(Dynamic Host Configuration Protocol)ソフトウェアエージェントは各デバイスにアドレス及びデフォルト名称を割り当て、前記アドレス及びデフォルト名称はサービスまたはデバイスのインタフェースに追加される。それから、ソフトウェアエージェント110は'部分的なXCEネットワーク(Network partial XCE)'を生成するために、ホームネットワークに局部的に接続された全てのデバイスからサブセットまたは'部分的なXCEデバイス(Device partial XCE)'を含むデバイスインタフェース72を集める。付加的な関連外部インタフェースが外部制御のための構造に追加される。例えば、'service\_id'は、ネットワークに接続されたデバイスのデバイスインタフェースによってソフトウェアエージェントから加入を含む受信された構造またはネットワークインタフェースライブラリ106内の名称/アドレスでありうる。その後、使用者はデータベースでサービスを検索でき、前記名称/アドレスを使用してライブラリの特定のデータ部分を含むアプリケーションをアクセスできる。従って、ネットワークは前記名称/アドレス情報により識別された多量の同じサービスを含むことができる。

'media'、これは例えば、チューナからの伝送ストリーム、PC DRAMからRAM、DCまたはDVDのためのディスク及びテープを含むメディアタイプのためのインタフェースを提供する。前記メディアは命名され確認され、制御器デバイスはネットワーク上で提供される一般のメディアを確認するXCEデータベースを検索できる。DVDディスクのような新たなメディアがネットワーク上で提供される時、ディスク上でプログラム資料を確認するデバイスインタフェース72の一部が適合に変換される。従って、全体デバイスインタフェース72が伝送される必要はなく、ただ適した部分がXCEデータベースに伝送される。アプリケーション信号の受信でライブラリソフトウェアエージェント110は新たな更新をフェッチでき、適した場所にそれを位置させられる。ディスクメディアの付加は、前記ネットワークにサービスを追加したりまたは前記ネットワークに他の装置を接続することと類似である。

'rate'、これは例えば、6Mbps/Secまたは19.2Mbps/Secのようにデバイスインタフェースのためのデータストリームレートに対する値を提供する。

'protocol'、これは前記データストリームのために使われたプロトコルを確認する。例えば、61883/1394またはU/1Pのように、一つ以上のプロトコルが提供されれば、望ましいプロトコルが選択される。

'stream\_format'、これはデジタルストリームオーディオ及びビデオ分割のためのパケットフォーマット及び/または圧縮標準を提供する。もし一つ以上のフォーマットが提供されれば、望ましいフォーマットがインタフェースメッセージを通じて選択される。制御器アプリケーション82は互換的なフォーマットがあるかどうかを決定できるフォーマットを調べられる。

'controls\_av'、これはA/Vメディア装置のためのメイン制御インタフェースを提供する。

'Flow\_control'、これは特定のデバイスのための方法であってPLAY、STOP、GOTO、RECORDのようなデータストリーム制御を提供する。その方法は、例えばPCソフトウェアを除いては埋め込まれた(Embedded)装置に対して変わらない。tkdrl制御は遅延された動作のために時間構成要素を含むことができる。

'Tuning'、これはチューニング制御のためのインタフェースを提供する。制御器サブデバイス14は、制御されたサブデバイス14のインタフェースに前述された電子プログラムガイド(EPG: Electronic Program Guide)データ構造の返信を求められる。

'Uicontrol'、これはディスプレイに対する輝度及びコントラスト、オーディオに対するボリューム及びベースのような修正を制御するために制御されたアプリケーション84に制御インタフェースを提供する。

'Timer\_record'、これは遅延された時間記録を具現するために制御器アプリケーション82に対してセットアップデータのためのインタフェースを提供する。直接的なチャンネル同調情報及びフロー制御(time\_aparams)情報が使われたりする。

【誤訂正38】

【訂正対象書類名】明細書

【訂正対象項目名】0102

【訂正方法】変更

【訂正の内容】

【0102】

他の観点で、本発明はネットワークでデバイス-デバイス命令及び制御のために存在する命令語具現の使用を提供する。デバイスは、内部対象及び存在する伝送メカニズムによって、実行時に2進数列を生成するAPIを有する。このような場合、ネットワークで一つのサブデバイス14からさらに他のサブデバイス14にXML遠隔手順呼び出し(XMLRPC: XML Remote Procedure Calls)を提供するために、存在するアプリケーションインタフェース具現はXMLサービスAPIに対する呼び出しに取り替えられる。従って、本来の具現はXMLサービスAPIのためのラップ(wrapper)と同等である。図18は、またXCE/XMLサービスAPIでラップに取り替えられたインタフェース具現として、点線内のCALまたはAV/Cのような他の命令語を使用して生成されたアプリケーションを示す。CAL命令語からXMLRPCフォーマットを変える例が下に示されている。

existing implementation:

```
void DeviceCALCommand(int command){
    ×
    ×
    /*
    create CAL formatted byte string to represent this
    object/method and output to the wire
    */
}
```

```

CreateCALFormattedByteString(command);/* different for
                                     every protocol*/
SendCALByteString(); /*different for every protocol
*/
}

-----
wrapping the XML Service API call:
void DeviceCALCommand(int command){
{
    /*
    replace CAL implementation with calls to the XML
ServiceAPI
    */
    CreateXMLMessage(command); /* always the same */
    sendXMLMessage(); /* always the same */
}

```

【誤訳訂正 39】

【訂正対象書類名】明細書

【訂正対象項目名】図面の簡単な説明

【訂正方法】変更

【訂正の内容】

【図面の簡単な説明】

【図 1】 本発明の第 1 観点に係るネットワークの具現のためのブロック図である。

【図 2】 デバイス制御及び通信概要の第 1 実施形態に係る図 1 のブロック図である。

【図 3】 多数のクライアントとサーバデバイスを含む本発明に係るホームネットワークシステムに対する第 1 実施形態を示すブロック図である。

【図 4】 図 3 のクライアントデバイス及びサーバデバイス具現の第 1 実施形態を示すブロック図である。

【図 5】 クライアントデバイス具現の第 1 実施形態を示す図面である。

【図 6】 サーバデバイス具現の第 1 実施形態を示す図面である。

【図 7】 相互間に通信及び制御を可能にするネットワークサーバデバイスの第 2 実施形態を示すブロック図である。

【図 8】 ネットワークで送信側のサーバデバイス、受信側のサーバデバイス及びクライアントデバイスを含むオーディオ/ビデオモデルの構造に対する第 1 実施形態を示すブロック図である。

【図 9】 オーディオ/ビデオモデルの他の実施形態を示す図面である。

【図 10】 ネットワークデバイスのための性能データテーブルの実施形態を示す図面である。

【図 11】 図 11 は、ネットワークデバイスのための特性データテーブルの実施形態を示す図面である。

【図 12】 ネットワークデバイス間で命令メッセージを発生させるビルディングブロックに対する第 1 実施形態の構成を示す図面である。

【図 13】 命令メッセージを発生させる図 12 に係るビルディングブロックの他の実施形態の構成を示す図面である。

【図 14】 ネットワーククライアント及びサーバデバイス間で相互作用の第 3 実施形態を示す図面である。

【図 15】 ネットワークデバイスインタフェースの API 拡張を定義する第 1 実施形態のブロック図である。

【図 16】 他のサーバデバイスのインタフェース説明文書をアクセスするサーバデ

バイスアプリケーションのための第1実施形態の構造を示す図面である。

【図17】 制御器サーバデバイスと制御されたサーバデバイスとの間の内部一デバイス制御構造の他の実施形態を示す図面である。

【図18】 ネットワークデバイス間のAPIレベルで通信スタックでウェブ標準共通ミドルウェア階層を提供するXMLプロトコルの具現を示す図面である。

【図19】 サーバデバイス間命令制御構造のまた他の具現を示す図面である。

【図20】 デバイスインタフェースライブラリとホームデバイスのための電化製品データベースとの関係を示す図面である。

【図21】 デバイスインタフェース定義に対する階層的形態の具現を示す図面である。

【図22】 図21のデバイスインタフェース定義で階層の第1実施形態を示す図面である。

【図23】 伝送器及び受信器デバイス間で共通伝送及び解釈手順を示す図面である。

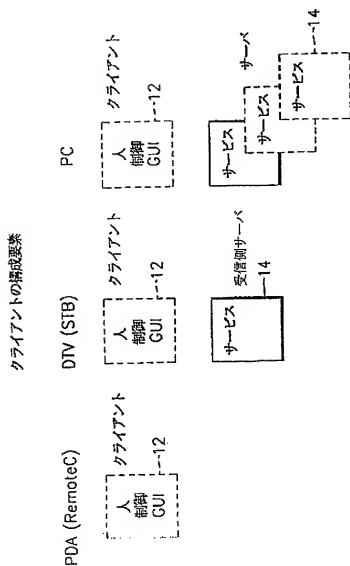
【図24】 本発明の第1観点に係る翻訳サービスを提供するためのパケットタイプとフォーマットの部分的なリストテーブルの第1実施形態を示す図面である。

【符号の説明】

- 10 ネットワーク
- 12 クライアントデバイス
- 14 サーバデバイス
- 16 通信リンク
- 18 GUIディスプレイ
- 20 サーバ制御プログラム
- 22 グラフィック制御対象(GCO) 使用者インタフェース説明書
- 24 クライアントデバイスレングラ
- 26 制御状態データ
- 28 クロック
- 30 命令語インタフェース
- 32 データストリーム送信側ハードウェア
- 34 データストリーム受信側ハードウェア
- 36 セッションマネージャ
- 40 サービスの機能的規格ブロック
- 42 メッセージ要素が構成されるブロック
- 44 産業標準化フォーマットブロック
- 46 メッセージ圧縮ブロック
- 48 メッセージ文字列構成ブロック
- 50 ソフトウェアクライアント制御プログラム
- 52, 54, 58, 60 ブロック
- 56, 62 ルックアップテーブル
- 64 ライブラリデバイス
- 66 通信スタック
- 68 XML層OUT
- 70 XML層IN
- 72 XMLインタフェース
- 74 XMLパーサ
- 76 XMLRPCコーデック
- 78 インタフェースフロッチャ
- 79 ホームネットワーク対象要請ブローカ
- 80 ホームネットワークインタフェースライブラリ
- 82 制御器アプリケーションプログラムコード

- 8 4 アプリケーションプログラムコード
  - 8 6 ホームネットワークデバイスウェブサーバ
  - 8 8 固有ルックアップテーブル
  - 9 0 ハンドラ
  - 9 2 ハードウェアサービス
  - 9 4 固有インタフェース
  - 9 6 第3デバイス
  - 9 8 ミドルウェア階層
  - 1 0 0 データベース
  - 1 0 2 CEデータベース
  - 1 0 4 XCEデータベース
  - 1 0 6 ホームネットワークインタフェースライブラリ
  - 1 0 8 製造会社手段
  - 1 1 2, 1 1 4, 1 1 6, 1 1 8 階層
  - 1 2 0 送信デバイス
  - 1 2 2 要請デバイス
  - 1 2 4 翻訳デバイス
- 【誤訳訂正40】
- 【訂正対象書類名】図面
- 【訂正対象項目名】図5
- 【訂正方法】変更
- 【訂正の内容】

【 図 5 】



【 誤 訳 訂 正 4 1 】

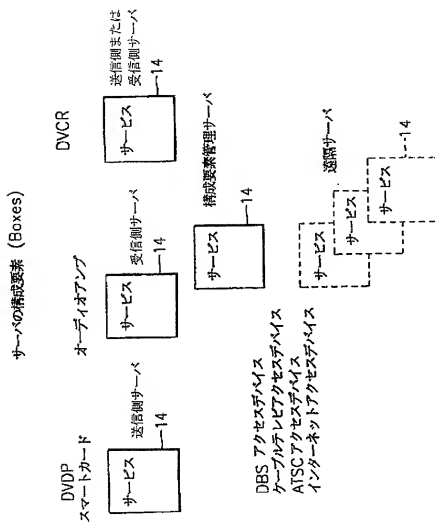
【 訂 正 対 象 書 類 名 】 図 面

【 訂 正 対 象 項 目 名 】 図 6

【 訂 正 方 法 】 変 更

【 訂 正 の 内 容 】

【図6】



【誤訳訂正 4 2】

【訂正対象書類名】図面

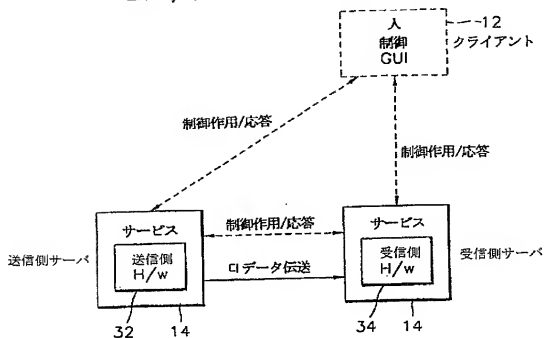
【訂正対象項目名】図 8

【訂正方法】変更

【訂正の内容】

【図 8】

基本 A/V、クライアント/サーバ/サーバ、モデル



【誤訳訂正 4 3】

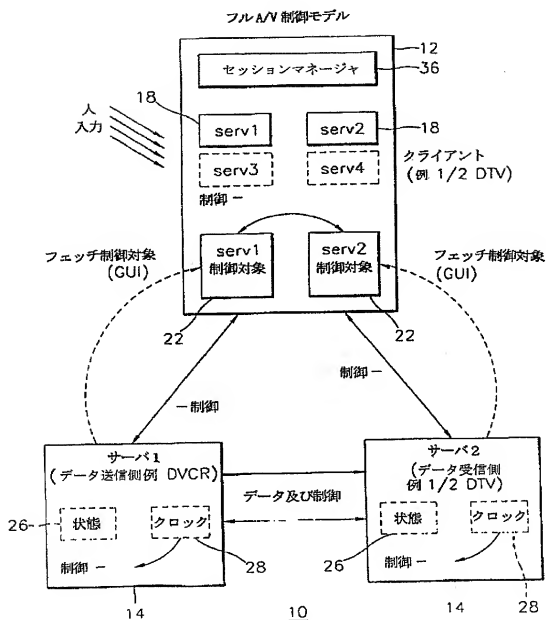
【訂正対象書類名】図面

【訂正対象項目名】図 9

【訂正方法】変更

【訂正の内容】

【図9】





## INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification <sup>6</sup> : <b>H04L</b>		A2	(11) International Publication Number: <b>WO 99/57837</b>
		(43) International Publication Date: 11 November 1999 (11.11.99)	
(21) International Application Number: <b>PCT/KR99/00221</b>		(81) Designated States: AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, CA, CH, CN, CU, CZ, DE, DK, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MD, MG, MK, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, UA, UG, UZ, VN, YU, ZW, ARIPO patent (GH, GM, KE, LS, MW, SD, SL, SZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).	
(22) International Filing Date: <b>7 May 1999 (07.05.99)</b>		Published <i>Without international search report and to be republished upon receipt of that report.</i>	
(30) Priority Data: 60/084,578 7 May 1998 (07.05.98) US			
(71) Applicant: SAMSUNG ELECTRONICS CO., LTD. [KR/KR]; 416, Maetan-dong, Paldal-gu, Suwon-city, Kyungki-do 442-373 (KR).			
(72) Inventors: HUMPLEMAN, Richard; 343, Lower Vintners Circle, Fremont, CA 94539 (US); WANG, Dongyan; 440 Oak Grove Drive #302, Santa Clara, CA 95054 (US).			
(74) Agent: LEE, Young, Pil; The Cheonghwa Building, 1571-18 Seocho-dong, Seocho-gu, Seoul 137-073 (KR).			
(54) Title: METHOD AND APPARATUS FOR UNIVERSALLY ACCESSIBLE COMMAND AND CONTROL INFORMATION IN A NETWORK			
(57) Abstract			
<p>A method and system for performing a service on a home network, by: connecting a first and a second home device to the home network; providing a database including a plurality of application interface description data objects, where each application interface description data object includes information in a structured format for commanding and controlling of a home device by one or more other home device connected to the network; the second home device accessing a first application interface description object for the first home device in the database; the first home device accessing a second application interface description object for the second home device in the database; sending control and command data from the first home device to the second home device utilizing the second application interface description object over the network; and sending control and command data from the second home device to the first home device utilizing the first application interface description object over the network. Whereby, the first and second home devices perform said service.</p>			



-2-

Home devices, such as home theater equipment, are often controlled using a single common control unit, namely a remote control device. This single common control unit allows a homeowner to control and command several different home devices using a single interface. Thus, may  
5 manufacturers have developed control units for controlling and commanding their home devices from a single interface.

One drawback associated with using the remote control unit to command and control home devices is that it provides static and command  
10 logic for controlling and commanding each home device. Another drawback associated with using remote control units is that known remote control units cannot control a plurality of diverse devices, and more particularly, cannot control a plurality of devices having different capabilities to communicate with each other in order to accomplish tasks or provide a service.

15 In conventional network systems a user provides commands using a remote control unit or device control panel. Once the user ceases, there is no controller unit or device in the network to provide commands for automatic operation. After a user initially controls and commands a first set of devices,  
20 conventional systems do not provide a mechanism for the first set of devices to automatically communicate with a second set of devices in the network as necessary in order to accomplish tasks without direct user control and command of the second set of devices. Further, conventional systems do not provide an efficient method for various network devices to obtain information  
25 about other network devices in the network for command and control.

There is, therefore, a need for a method and a system which provides dynamic control and command devices in a home network. There is also a need for such a method and system to provide the ability to control a plurality  
30 of diverse devices having different capabilities to communicate with each other in order to accomplish tasks or provide a service. There is also a need for such a method and system to provide the ability for various network

-3-

devices to automatically command and control other various network devices. There is also a need for such a method and system to provide universally accessible command and control information for inter-device communication.

5

### Summary of the Invention

The present invention satisfies these needs. In one embodiment the present invention provides a method and system for performing a service on a home network, by: connecting a first and a second home device to the home network; providing a database including a plurality of application interface description data objects, where each application interface description data object includes information in a structured format for commanding and controlling of a home device by one or more other home devices connected to the network; the second home device accessing a first application interface description object for the first home device in the database; the first home device accessing a second application interface description object for the second home device in the database; sending control and command data from the first home device to the second home device utilizing the second application interface description object over the network; and sending control and command data from the second home device to the first home device utilizing the first application interface description object over the network. Whereby, the first and second home devices perform said service.

In one version of the invention, the first home device stores first application interface data therein, and the second home device stores second application interface data therein. The database is formed by querying the first and second home devices to transfer said application interface data for the first and second home devices to the database device. The database can be stored in a database device and connected to the network for universal access by network devices. In that case, the first application interface description object for the first home device can be provided from the database to the second home device over the network. Further, the second

-4-

application interface description object can be provided from the data base to the first home device over the network.

Further, three or more home devices can be connected to the network,  
5 wherein at least one home device accesses the database to query the application interface description objects of a plurality of home devices for sending command and control data to the plurality of home devices over the network. Each application interface description object can include data in a structured format. The structured format can include XML format.

10

### **Brief Description of the Drawings**

These and other features, aspects and advantages of the present invention will become better understood with regard to the following description, appended claims and accompanying drawings where:

15 Figure 1 shows a block diagram on an embodiment of a network according to one aspect of the present invention;

Figure 2 shows the block diagram of Figure 1 in an example device control and communication scenario;

Figure 3 shows a block diagram of an example home network system  
20 according to the present invention which includes a plurality of client and server devices;

Figure 4 shows a block diagram of example embodiments of a client device and a server device of Figure 3;

Figure 5 shows example embodiments of client devices;

25 Figure 6 shows example embodiments of server devices;

Figure 7 shows a block diagram of two example networked server devices capable of communication with, and control of, one another;

Figure 8 shows a block diagram of an example architecture of an audio/video (A/V) model including examples of a source server device, a sink  
30 server device and a client device in a network;

Figure 9 shows another example audio/video (A/V) model;

Figure 10 shows an example capabilities data table for a network

device;

Figure 11 shows an example attribute data table for a network device;

Figure 12 shows an example configuration of building blocks for generating command messages among networked devices;

5        Figure 13 shows another example configuration of the building blocks of Figure 12 for generating command messages;

Figure 14 shows three examples of interaction among networked client and server devices;

10       Figure 15 shows an example block diagram for definitions of API extensions of networked device interfaces;

Figure 16 shows an example architecture for a server device application accessing the interface description document of another server device;

15       Figure 17 shows another example inter-device control architecture between a controller server device and a controlled server device;

Figure 18 shows an embodiment of an XML protocol providing a Web standard common middleware layer in a communication stack at the API level between networked devices;

20       Figure 19 shows another embodiment of server device to server device command and control architecture;

Figure 20 shows the relationship between a device interface library and consumer electronics definition data base for home devices;

Figure 21 shows hierarchal form of an embodiment of a device interface definition;

25       Figure 22 shows an example of layers in device interface definition of Figure 21;

Figure 23 shows a command transmission and interpretation process between a sender and receiver device; and

30       Figure 24 shows an example table of a partial list of packet types and formats for providing translation services according to an aspect of the present invention.

### Detailed Description of the Invention

In one aspect, the present invention provides inter-device communication in a network such as a home network. As home devices become more intelligent and can share information, inter-device communication allows a user to interconnect devices in a network to take advantage of the information sharing capabilities of those devices. As such, inter-device communication plays a crucial role in affording a user with the ability to fully and flexibly utilize the networked devices.

Referring to Figure 1, in an embodiment of the present invention, a network 10 includes at least one client device 12 and at least one server device 14 interconnected via a communication link 16. The communication link 16 can include a 1394 serial bus providing a physical layer (medium) for sending and receiving data between the various connected home devices. The 1394 serial bus supports both time-multiplexed audio/video (AVV) streams and standard IP (Internet Protocol) communications. In certain embodiments, a home network uses an IP network layer as the communication layer for the home network. However, other communication protocols could be used to provide communication for the home network.

Each client device 12 may communicate with one or more server devices 14 in the network 10. Further, each server device 14 may communicate with one or more other server devices 14, and one or more client devices 12, in the network 10. Each client device 12 can include a user communication interface including input devices such as a mouse and keyboard for receiving user input, and a display for providing a control user interface for a user to interact with the networked devices. The user interface can include a graphical user interface (GUI) display 18 for providing information to the user. Referring to Figure 2, as defined herein, each server device 14 provides a service for the user, except control user interface, and each client device 12 provides control user interface for user interaction with the network 10. As such, only client devices 12 interact directly with users,

-7-

and server devices 14 interact only with client devices 12 and other server devices 14. Example services can include MPEG sourcing/sinking and display services.

5           Figure 3 shows a block diagram of an example home network 10 that includes a plurality of client devices 12 and a plurality of server devices 14. Each server device 14 may include hardware as a resource in the network for providing services to the user. Further, each server device 14 may store a server or service control program 20 for controlling the server hardware, and  
10          include a graphical control object (GCO) user interface description 22 for user interface with the server control program 20 as shown in Figure 4.

          For control between a controlling client device 12 and a controlled server device 14, the client device 12 accesses the GCO 22 of the server  
15          device 14 by, for example, transferring the GCO 22 from the server device 14 to the client device 12 over the network. The client device 12 then uses the transferred GCO 22 to create a control user interface GUI 18 for the user to communicate with the control program 20 of the server device 14 from the client device 12 over the network. The user provides command and control to  
20          at least the control program 20 of the server device 14 from the client device 12.

          Storing the GCO 22 of each server device 14 in the server device itself may reduce the processing and storage requirements of the client devices 12  
25          in networks with several server devices 14. Further, storing the GCOs 22 in the server devices 14 may allow each server device 14 to provide its own GUI look and feel to the user, and allows for modification or updating of the GCOs 22 without modifications to client devices 12.

30          Referring to Figure 4, to provide command and control between a client device 12 and the server device 14, in one embodiment, the client device 12 can include a renderer 24 for displaying a GUI 18 using a GCO 22

-8-

stored in the client device 12 or transferred to the client device 12 over the network from a desired server device 14. For example, in an initial device selection phase, the client device 12 can fetch the GCO 22 of at least one server device 14 over the network, and the renderer 24 displays a GUI 18 using the GCO 22 for controlling the server device 14. Preferably, the GUI 18 is customized to the server device 14 and can include a built-in command set for controlling the server device 14.

In addition, the GUIs 18 of various server devices 14 may include commonalities such as: (1) a common GCO model type for the client device renderer 24 to display GUIs 18, (2) common communication protocols for transferring the GCOs 22 from various server devices 14 to the client device 12, and (3) common communication protocols for GUI interaction from the client device 12 to the control program 20 of the corresponding server device 14, wherein the client device 12 does not require a built-in knowledge of a particular server device 14 being controlled.

Referring still to Figure 4, a server device 14 may include one or more server control programs 20 to control the server hardware for providing a service. The GUI interface 18 from the GCO 22 of the server device 14 provides interface to the server device control programs 20. The server device 14 may also include control state data 26 indicating the control status of the server device 14 and server device hardware in providing a requested service.

For example, the control state data 26 can include the status of control information in the GUI 18 for the server device 14, such as timer setup for a recording action in a VCR server device. The control state data 26 is stored in the controlled server device 14, and displayed to a user through the GUI 18 of the server device 14 at the controlling client device 12, for user control of the server device 14. Preferably, the controlling client device 12 for displaying the GUI 18 of the server device 14 does not retain knowledge of the control

-9-

state data 26 for the controlled server device 14.

Each server device 14 can be controlled by one or more client devices 12. As such, the control state data 26 stored in the server device 14 includes status of the information in the GUI 18 of the server device 14 at each of the controlling client devices 12. For example, when the user controls a server device 14 using a first client device 12, upon completion of the user control, the information in the GUI 18 of the server device 14 at the first client device 12 is saved by the server device 14 in the control state data 26 of the server device 14.

Alternatively, while the user is interacting with the GUI 18 of the server device 14 at the first client device 12, the control state data 26 of the server device 14 is updated with the information in the GUI 18 of the server device 14 at the first client device 12, and upon completion of user control, the control state data 26 is retained in the server device 14. When the user controls the server device 14 using a second client device 12, the control state data 26 is made available to the user via the GUI 18 of the server device 14 at the second client device 12 for further control. The user can also use the first client device 12 at a later time to control the server device 14, whereupon the control state data 26 is made available to the user via the GUI 18 of the server device 14 at the first client device 12 for further control. The server device 14 can also include a clock 28, or maintains the current time, to allow time delay action based on time or clock input from a user, as described below.

A client device 12 and a server device 14 can be physically bundled together as one unit such as a DTV. In that case, the server device 14 includes a control program 20 for controlling the server hardware, and the client device 12 provides control user interface to the server control program 20 for control and command of at least the server hardware. Figure 5 shows examples of client devices 12 that may include: (1) a PDA(RemoteC) for

-10-

displaying a GUI, (2) a DTV(STB) for displaying a GUI and including a sink server comprising audio and/or video program stream destination server, and (3) a PC for displaying a GUI and including at least one server device for providing multiple services. Hardware and executables in a DTV or PC client device can also be controlled by other client devices. Figure 6 shows example server devices 14, including: (1) a DVDP SmartCard as a source server device, (2) an Audio Amplifier as a sink server device, (3) a DVCR as either a source or a sink server device, and (4) a Management Server for managing remote server devices. The Management Server can be included in a DBS-STB, Cable TV-STB, or ATSC-STB, for example. Such devices include a Management Server for local control or management of the internal workings of the STB. Further, external servers accessed through an external network can be utilized by local client devices for services such as Video-on-Demand, Enhanced-TV, and Internet commerce, for example.

Referring to Figure 7, communication and control between two server devices 14 is accomplished by the control programs 20 of the server devices 14 communicating command and control data therebetween. A server device 14 can control one or more other server devices 14 over the network. And, a server device 14 can be controlled by one or more server devices 14, and by one or more client devices 12. Further, a user can utilize a client device 12 to control and command a first set of server devices 14, and the first set of server devices 14 can automatically command and control a second set of server devices 14 without user involvement, as necessary to perform services to the user.

For example, for automatic time-delay operation, a user can "log on" to a client device 12 to control a first set of server devices 14 and specify desired services. The user then "logs off" from the client device 12. The first set of server devices 14 perform communication and control among themselves, and at a later time, one or more of server devices 14 in the first set automatically control a second set of server devices 14 as necessary to

-11-

collectively provide the desired services without user involvement.

Figure 7 shows example embodiments of two server devices 14 capable of communication with, and control of, one another. Each server device 14 includes a control program 20, a clock 28 and control state data 26 described above. Each server device 14 can also include a GCO 22 for the server device 14 to be directly controlled by a client device 12. However, a GCO 22 does not need to be included in a server device 14 that is not directly controlled by a client device 12 and only communicates with other server devices 14. Each server device 14 also includes a command language (CL) interface 30 and a library of commands. The library of commands includes the commands that the server device 14 utilizes to send and receive information for providing its service. However, a command language is not necessary for user control as shown in Figure 4 and described above.

Figure 8 shows an example audio/video (A/V) model including a source server device 14, a sink server device 14 and a client device 12 in the network. The source server device 14 includes a control program 20 for controlling data stream source hardware 32 of the source server device 14, and the sink server device 14 includes a control program 20 for controlling data stream sink hardware 34 of the sink server device 14. In an example operation, a user utilizes the client device 12 to control the source server device 14 to start the data stream source hardware 32, and to control the sink server device 14 to start the data stream sink hardware 34. Upon initiation of data transfer from the data stream source hardware 32 to the data stream sink hardware 34, the user can relinquish the client device 12. Alternatively, the user can program the initiation of the data transfer for a future time and relinquish the client device 12. Thereafter, the data stream source hardware 32 of the source server device 14 and the data stream sink hardware 34 of the sink server device 14 automatically initiate the data transfer at the time programmed by the user.

-12-

For example, the data stream source hardware 32 can include a Tuner-Access Device such as a Direct Broadcast Satellite (DBS). A DBS is a multi-channel alternative to cable television, and provides cable-like television programming directly from satellites on small (18 inch to 3-foot diameter) satellite dishes. With DBS, several standard analog television signals are digitally compressed onto a single satellite transponder thereby allowing up to 200 or more channels receivable with a dish pointed at a fixed position in the sky. The data stream sink hardware 34 can include a Digital Video Cassette Recorder (DVCR) which comprises a digital VCR that is able to decode compressed digital video signals on playback. The user provides command and control data including "time-delay record" event data for the DVCR and a "time-delay select a program" event data for the Tuner-Access Device. After the time-delay, the Tuner-Access Device selects the desired program, and sources program data to the DVCR which receives and records the program data without further control actions from the user.

Figure 9 shows another example audio/video (A/V) model including at least a source server device 14 SERVER1, a sink server device 14 SERVER2 and a client device 12 in the network 10. The client device 12 includes a session manager 36 with a user interface for displaying selection information for a user to select and control the server devices 14 SERVER1, SERVER2 and other server devices 14 such as SERVER3 and SERVER4 (not shown). The selection information can include iconic symbols designated as Serv1, Serv2, Serv3 and Serv4 in the session manager 36 for a user to select the server devices 14 SERVER1, SERVER2, SERVER3 and SERVER4, respectively. The source server device 14 SERVER1 can include a DVCR and the sink server device 14 SERVER2 can include a 1/2DTV.

In one example operation, upon selection of the server devices 14 SERVER1 and SERVER2, the client device 12 transfers the GCO 22 of each server device 14 to the client device and displays a corresponding GUI 18 for each of the server devices 14 SERVER1 and SERVER2. The user can

-13-

interact with the GUI 18 of each server device 14 to provide command and control to the corresponding server device 14 for service. Each server device 14 can provide service alone or in combination with other server devices 14. Further, the session manager 36 transfers control state data 26 between the  
5 GUIs 18 of the server devices 14 in the client device 12 as necessary for the corresponding server devices 14 to perform a service. Based on the user command and control information, two or more of the server devices 14 can communicate command and control information therebetween to provide a user requested service.

10

The session manager 36 can include a software agent which functions to access and display available home network services provided by various server devices 14 in the network 10. The software agent can additionally match the capabilities of various server devices 14 in the network 10 and  
15 display selection information for only those server devices 14 that have compatible capabilities. Further, the session manager 36 can match the selections made in the GUI 18 of one server device 14 to the selections in GUI 18 of another server device 18 to help the user provide meaningful command and control information to the server devices 14.

20

In another example operation, the session manager 36 executes the software agent which searches the network and discovers the server devices 14 connected to the network. The software agent also accesses capabilities data stored in each server device 14 to determine the capabilities of the  
25 server devices 14 and provide information about those capabilities to the user. The session manager 36 then displays the selection icons Serv1, Serv2, Serv3 and Serv4 for the server devices SERVER1, SERVER2, SERVER3 and SERVER 4 as shown in Figure 9.

30

The session manager 36 initially enables all the selection icons Serv1, Serv2, Serv3 and Serv4 to allow the user to select from among all four selection icons. After the user selects the server device SERVER1 by clicking

-14-

on the Serv1 selection icon, the session manager 36 determines that the server devices SERVER3 and SERVER 4 are incompatible in capability with the server device SERVER1. As such, the session manager 36 disables the selection icons Serv3 and Serv 4 for server devices SERVER3 and  
5 SERVER4, respectively. The user can then click on the icon Serv2 to command and control the server device SERVER2.

As the user interacts with the GUI 18 of a selected server device 14, control and command information input by the user into each GUI 18 provide  
10 additional capabilities information which affect further server device selections by the user. For example, if a VCR server device 14 is selected, further action by the session manager 36 in enabling or disabling selection icons for other server devices 14 is affected by a user decision to play or record.

Each server device 14 in the network has one or more service capabilities as discussed above by way of example with reference to the server devices in Figure 9. Each service capability includes sourcing or sinking of information. For example, a TV has the sinking capability of receiving video and audio streams, a VCR device can source (transmit) and  
20 sink (receive) video and audio signals, and a PC may be able to transmit and receive video, audio and data. Each sourcing capability has a complementing, and compatible, sinking capability. Similarly, each sinking capability has a complementing, and compatible, sourcing capability. For example, a video output capability of one device is complemented by a video  
25 input capability of another device.

Since each device 14 can be a source or sink for several different services on the network, each device 14 stores a capabilities data table (Capabilities Table 1) as shown by example in Figure 10. The first column of  
30 Table 1 identifies the service capabilities of a device 14, and the second column identifies whether the device 14 is a source or a sink for a corresponding service in the first column. Using the capabilities data Table 1,

-15-

new services can be implemented while maintaining compatibility with older devices. For example, if a new service is developed that is compatible with an older service, both the new and the old service can be entered into the capabilities data Table 1 for a device implementing the new service, whereby  
5 the implementing device remains compatible with older devices using the old service.

In one implementation, a Device Manager conducts a matching or comparison of device source and sink services. For example, the Device  
10 Manager can be implemented as a software agent to compare the capabilities or properties of various devices 14 and locate devices 14 with matching capabilities. For example, in a case where the service is a media stream from a first device 14 across the network to a second device 14, the Device  
Manager compares the capabilities of the first and second devices 14 to  
15 assist the user in making a sensible selection of the second device 14 which is compatible with the capabilities of the first device 14. The following is an example list of service capabilities for an embodiment of a server device 14:

20           Stream\_format\_video\_dv  
            Stream\_format\_video\_mpeg2tpt  
            Stream\_format\_video\_dsstpt  
            Stream\_format\_video\_mpeg2pes  
            Stream\_format\_video\_mpeg210801-tpt

Each device 14 can further store an attribute data table (Attribute  
25 Table 2) including pertinent attributes of the device, shown by example in Figure 11. A name and a value define each attribute within Table 2. Though character lengths are shown in Table 2, they are not required. The attribute data is available to other devices 14 on the network 10 to facilitate interoperability and to store device information. For example, a Device Page  
30 as described below uses the Attribute Table 2 to store the device name. Other fields can be added to the attribute data Table 2 as necessary.

In the user-to-client device control model described above, attribute data can be displayed on the GUI page of the server device 14 at the client

-16-

device 12. Alternatively, a second level device information home page can be utilized to display said attribute data. Further, the attribute data in the form of a text or Extensible Markup Language (XML) file can be accessed by a software agent. For the device-to-device control model, the attribute data for the controlled device is stored in the device interface application interface.

The Device Location attribute field in the Attribute Table 2 is used to store the location or group for each device 14. The Device Type attribute field specifies the device type, such as VCR, DVD, DTV, Camcorder, PC, Security System, etc. for the particular device 14. The Device Type attributes field is used to select a default device icon to represent the device within the Device Page if the device itself does not supply one. The Attribute Table 2 can include multiple entries for the Default Source and the Default Sink attributes fields. Each such entry represents a different default source or sink device 14 for each data type handled by the device 14.

Preferably, the capabilities and attributes data are packaged into structured data using a hierarchical language. This provides a common method of retrieving the capabilities and attributes data that are used for other purposes such as in GCO transfer and server device-to-server device control. As an example, the attributes data can include the following structured data format:

```

25      <DEVICEATTRIBUTES>
          <ATTRIBUTE name=DeviceManufacturer value="Samsung
Inc.">
          <ATTRIBUTE name=Manufacturer URL
value=www.samsung.com>
          <ATTRIBUTE name=ManufacturerIcon value="logo.gif">
30      <ATTRIBUTE name=DeviceName value="Samsung DSS">
          <ATTRIBUTE name=DeviceModel value="SCH1900">
          <ATTRIBUTE name=DeviceType value=DDS>
          <ATTRIBUTE name=DeviceLocation value="Livingroom">
          <ATTRIBUTE name=DeviceIcon value="device.gif">
35      <ATTRIBUTE name=DeviceAddress value=105.144.30.17>
      </DEVICEATTRIBUTES>

```

-17-

As an example, the capabilities data can include the following structured format:

```
5      <DEVICECAPABILITIES>
        <CAPABILITY type=MPEG2 value=Source>
        <CAPABILITY type=MPEG2 value=Sink>
        <CAPABILITY type=MPEG3 value=Source>
        <CAPABILITY type=MPEG3 value=Sink>
10     </DEVICECAPABILITIES>
```

10 An application interface language is utilized to allow different server devices 14 to perform device-to-device control, including sever device-to-server device control. The application interface language includes command languages, and can be described using XML, as detailed below. The control  
15 program 20 of one server device 14 remotely controls the control program 20 of another server device 14 over the network, without using GUIs 18 or user involvement. An example of device-to-device control is automatic operation. A user initially provides control through a client device 12 for a desired service, and subsequently two or more server devices 14 automatically  
20 communicate and control one another without further user interaction to provide the service.

Referring to Figures 12 and 13, preferably a standard application interface language is utilized to allow interoperability among various control  
25 programs 20 in various server devices 14. In one embodiment, the standard application interface language includes the following building blocks: (1) functional specification of service 40 such as in a service function database, (2) a block where elements of a message are composed 42, (3) industry standard format 44, (4) message compression 46, and (5) message string  
30 construction 48 to output structured message data.

Figure 12 shows an example configuration of the building blocks to perform the function of generating command messages. Each message item is composed from the functional specification of service and standardized by  
35 selecting an industry standardized compressed form (Hex) label for the

-18-

message item. A group of such message items are assembled to create a complete command string. Existing command languages such as CAL and AV/C operate as shown in Figure 12. However, such command language mechanisms specify binary or hex code messages and system operation on physical devices on the physical interface, and are based on hardware specifications. Therefore, such command languages may be less desirable for a network layer based control mechanism where a control system specification includes naming, addressing, device capability discovery, communication language and command messages at the application level software level, where one software application program 20 in a controller device 14 locates and controls another software application 20 program in a controlled device 14 over the network 10. Said control mechanism is more suitable for devices such as digital appliances including appliances (e.g., DVCR) as well as multi-purpose, multi-application devices such as computers.

Figure 13 shows a preferred example configuration of the building blocks of Figure 12 to perform the function of generating command messages. In Figure 13, the positions of the industry standard format 44 and the message compression 46 are different than in Figure 12. A number of textual standard forms are selected from the functional specification service 40 to make a complete message. Later the message may be compressed by a lower layer of the protocol stack. Figure 13 represents a method of performing service or device command and control for consumer electronics (CE). Message composition can be defined by the XML standard syntax and compression can be performed by another protocol layer such as HTTP. A command interface language is utilized at the application software 20 interface level, rather than lower hardware levels. As such, the network protocol stack is governed by commands in said language, and each of a controller device 14 and controlled device 14 can be viewed as integrated components of the network for message transmission therebetween.

-19-

Referring to Figure 14, three different instances of interaction among client devices 12 and server devices 14 are shown. In the first instance "A", a human user communicates with a remote service application "S". The user utilizes a browser in a client device 12 as the user interface, wherein the browser controls service programs 20 in the service application "S" and receives response in Hyper Text Markup Language (HTML) or XML formats. A secondary server is included with the browser to accept XML based asynchronous command message postings. For example, for a DVCR the secondary server 14 can accept command messages such as "VCR FAILED: TAPE BROKE." A software agent including a browser can be utilized to display the command messages for a user in the browser's GUI for later attention by the user and control of the DVCR. Preferably, an XML based client device 12 includes an HTTP1.1 server capability to respond to command initiated elsewhere for server device to server device command and control.

In the second instance "B", the user is replaced by a software client control program 50. The software client control program 50 generates XML based command postings to the service application "S" and receives back XML command postings. And, in the third instance "C", the software client control program 50 is replaced by an application such a server device control program 20, wherein commands and responses are exchanged between two service applications 20. In that regard, instance "B" is a special case of instance "C" with a null service.

An application interface language based on XML is used for control between a first server device 14 and a second sever device 14 (device-to-device or service-to-service) for devices or services that are world wide web (Web) enabled and Internet enabled. The application interface language is based on the Web standard, middleware layer. In one embodiment, device-to-device control includes remotely controlling the control program 20 or Application, in one server device 14 from another server device 14 in the

-20-

network 10. As such, the interfaces (API) to such Applications 20 are made available over the network using API extensions. Preferably, the API extensions utilize a standard format, such as an XML-based interface, to provide overall interoperability.

5

Referring now Figure 15, there is shown block diagram definitions of API extensions for a first *Application A*, designated as *Service A*, and a second *Application B*, designated as *Service B*, communicating over the network 10. For example, the *Service A* can be the control program for a first server device *A* in the network, and the *Service B* can be the control program for a second server device *B* in the network. The server device *B* sends commands to the server device *A*. For this example, the first and second service devices *A* and *B* can include CE devices.

Referring to the API extensions for the *Service A*, the first upper-most block 52 provides a comprehensive definition or data base of CE objects and methods using English words to describe CE devices. The comprehensive definition or data base can also be in C, XML or other formats capable of representing objects and their respective methods. The comprehensive definition or data base utilizing XML is termed XCE definition. The second block 54 provides a format for representation of an API in XML form for all devices 14, designated as an interface data type definition INTERFACE.DTD.

A software agent, designated as *Tool A*, utilizes a subset of the XCE definition for *Service A*, and uses the interface data type INTERFACE.DTD for *Service A* to create an XML form document, INTERFACE-A.XML. The document INTERFACE-A.XML describes the objects and methods supported by the *Service A* according to the document type definition INTERFACE.DTD for *Service A*. Other data type definitions can also be used to create the INTERFACE-A.XML document.

The software *Tool A* also creates a look-up table 56 to convert from

XML messages from *Service B* on the network interface, to API definitions for *Service A*, programmed in C for example, and complied to executable binary. Preferably, the look-up table 56 is created at compile time, whereby during run-time, incoming XML form method messages (commands) from *Service B* are converted to the API format created by the complied application C code for *Service A*. The look-up 56 table provides run-time translation of XML object method calls from *Service B* into device native language calls for *Service A*. The look-up table 56 is complied with the device control program 20 for local execution on the server device *A* for *Service A*.

10

The INTERFACE-A.XML can be used by *Service A* for validity checks if it encounters an error in a received message. The INTERFACE-A.XML can also be used by a foreign Application such as *Service B* to determine the message format for *Service A* before communicating with *Service A*. Further, if a message from *Service B* to *Service A* causes an error, *Service B* can access the INTERFACE-A.XML document to diagnose the error.

15

Referring to the API extensions for the *Service B*, the first block 58 provides a comprehensive definition or data base of CE objects such as the XCE definition for *Service A* above. The next block 60 provides a language definition for making XML form method (command) calls to remote API services or devices such as the API for *Service A*. The language definition is a document type definition Method Request CALL.DTD which describes interaction with objects on the network.

20

A software agent, designated as *Tool B*, utilizes at least a subset of the objects and methods in the XCE definition for *Service B* and the CALL.DTD document, to generate a look-up table 62 for converting commands from a complied C program code for *Service B* into XML form method requests. As such, the look-up table 62 provides conversion between a method invoked by *Service B* (e.g., "PLAY") and the XML document or message that carries the method call across the network interface to *Service A*, for example. The

25

30

-22-

subset of the XCE definition used by software *Tool B* depends on the extent and nature of use of the network. For example, the subset can be selected to provide global or restricted use of all available services on a home network.

- 5 Therefore, the API extensions provide for communication between various devices on the network using XML. In the example above, the program code 20 for *Service B* generates method calls to an API, and the API calls are converted to XML form to comply with the Web/Internet standard XML for inter-device communication. The XML method calls (messages) are
- 10 sent to *Service A* over the network, and *Service A* reconverts the XML method calls from the network interface to program code API definitions for *Service A*. This conversion and re-conversion provides Web/Internet compatibility for diverse devices in the network with program code APIs which would otherwise require binary compatibility between different devices.
- 15 Examples of the XML interface blocks utilizing the block diagrams in of Figure 15 are shown below.

```

-----
20      interface.dtd
      rules for describing an object interface in xml
-----

25      <!ELEMENT      paramater      #PCDATA>
      <!ATTLIST      paramater
                Type      CDATA #REQUIRED
                >

30      <!ELEMENT      method      (#PCDATA, (parameter)+)>
      <!ELEMENT      method      (#PCDATA, (method)+)>

35      -----
      interface.h
      example object interface in c
      -----

40      /* object */
      typedef struct Stream {
                int id;

```

-23-

```

};

/* methods */

5 void StreamPlay (int id, int speed);
  void StreamStop (int id);

-----
10 interface.xml
   the same object in xml using rules of interface.dtd
-----

15 <object>Stream
    <method>Play
        <parameter type="int">id</parameter>
        <parameter type="int">speed</parameter>
    </method>
    <method>Stop
        <parameter type="int">id</parameter>
    </method>
    </object>

25 -----
   call.dtd
   rules for describing a c function call in xml
-----

30 <!ELEMENT    parameter    #PCDATA>
   <!ATTLIST   parameter    parameter
        value          CDATA #REQUIRED
        >

35 <!ELEMENT    method        {#PCDATA, (parameter)+}>
   <!ELEMENT    object        {#PCDATA}>
   <!ELEMENT    call          (object,method)>

40 -----
   controller.c
   example controller command in c
-----

45 .
   .
   .
   StreamPlay (0x01ae, 500);
   .
50 .
   .
-----

55 call.xml

```

-24-

the same command in xml using call.dtd

```

5 <!-- example to play a stream -->

    <call>
      <object>Stream</object>
      <method>Play</method>
10   parameter value="500">speed</parameter>
    </call>

```

Further, the above provides examples of interface definitions  
 INTERFACE.DTD and CALL.DTD used to create description documents of  
 15 available services, INTERFACE.XML, described above. The CALL.DTD  
 definition includes a rule set for generating method call or function call  
 message such as XML Remote Procedure Call (RPC) or XMLRPC  
 messages. The CALL.DTD definition describes an output interface of a  
 controller service 14. In a home network, for example, INTERFACE.XML  
 20 represents the services available on the home network. The available  
 services are a subset of the entire services in the CE space.

In a One-Touch-Record (OTR) scenario, a user is in control of a  
 Tuner-Access-Device such as a Satellite STB. The user controls the tuning  
 25 using an Electronic Program Guide (EPG) such as a graphical user interface  
 representation of program listings. OTR record provides the user with a  
 service including selection of a future program from the EPG for recording  
 without the user accessing the VCR graphical user interface to program the  
 VCR for a Time Delayed Recording. OTR automates the control of the VCR.  
 30 Below is an example control list of actions in OTR.XML: (1) StreamOpen =  
 play the selected program stream output to the network from a Satellite STB;  
 for OTR this control is local to the STB device; (2) StorageOpen = open a  
 storage service; and (3) StorageRecord = Send the Record command across  
 the network to the VCR.

call.dtd

-25-

rules for describing a c function call in xml

```

5  <!ELEMENT    parameter    #PCDATA>
   <!ATTLIST   parameter
       value    CDATA #REQUIRED
       >

10 <!ELEMENT    method        (#PCDATA, (parameter)+)>
   <!ELEMENT    object        (#PCDATA)>
   <!ELEMENT    call          (object,method)>

```

interface.dtd

example for describing an object interface in xml

```

20 <!ELEMENT    parameter    #PCDATA>
   <!ATTLIST   parameter
       value    CDATA #REQUIRED
       >

25 <!ELEMENT    method        (#PCDATA, (parameter)+)>
   <!ELEMENT    object        (#PCDATA, method+)>

```

interface.xml

```

30      this document describes various CE services offered -a
      subset of the whole CE space.

```

```

35 <?xml version="1.0"?>
   <!DOCTYPE interface SYSTEM "interface.dtd">

   <object>Stream
     <method>Open
40       <parameter type="int">id</parameter>
       <parameter type="int">channel</parameter>
     </method>
     <method>Close
       <parameter type="int">id</parameter>
     </method>
45 </object>
   <object>Control
     <method>Set
       <parameter type="int">id</parameter>
       <parameter type="int">level</parameter>
50     </method>
   </object>

   <object>Storage
     <method>Open
55       <parameter type="int">id</parameter>

```

-26-

```

        <parameter type="int">channel</parameter>
    </method>
    <method>Record
        <parameter type="int">id</parameter>
5    </method>
    <method>Play
        <parameter type="int">id</parameter>
        <parameter type="int">speed</parameter>
10    </method>
    <method>Stop
        <parameter type="int">id</parameter>
    </method>
    <method>Close
        <parameter type="int">id</parameter>
15    </method>
</object>

<object>Display
    <method>Open
20    <parameter type="int">id</parameter>
        <parameter type="int">channel</parameter>
    </method>
    <method>Render
        <parameter type="int">id</parameter>
25    </method>
    <method>Blank
        <parameter type="int">id</parameter>
    </method>
    <method>Control
30    <parameter type="int">id</parameter>
        <parameter type="int">cid</parameter>
        <parameter type="int">level</parameter>
    </method>
    <method>Close
35    <parameter type="int">id</parameter>
    </method>
</object>

40    <!-------
        otr.xml

        an xml representation of one touch record

45    c representation:

        StreamOpen (100,2);/* play a stream (pushed by satellite feed
*/
        StorageOpen (24,2);/* open a storage service */
50    StorageRecord (24);/* record the stream */

        -----

55    <?xml version="1.0"?>
        <!DOCTYPE interface SYSTEM "call.dtd">

```

-27-

```

    <call>
      <object>Stream</object>
      <method>Open</method>
        <parameter value="100">id</parameter>
        <parameter value="2">channel</parameter>
5    </call>
    <call>
      <object>Storage</object>
      <method>Open</method>
10    <parameter value="100">id</parameter>
      <parameter value="2">channel</parameter>
    </call>
    <call>
      <object>Storage</object>
      <method>Record</method>
15    <parameter value="100">id</parameter>
    </call>

```

As discussed above in relation to Figure 15, a first device *B* can

20 access the INTERFACE.XML document of a second device *A* to examine the device capabilities and API interface details of the second device *A* and determine supported functionality and command details of the second device *A*. In particular, the first device *B* can determine overlapping, and therefore useable, methods supported by first device *B* and the second device *A*.

25 Figure 16 shows an example wherein a first server device *B* including an *Application B* accesses the INTERFACE-A.XML document of a second server device *A* including an *Application A*. The first server device *B* includes a INTERFACE-B.XML document for comparison with that of a INTERFACE-A.XML document in the second server device *A*.

30 In one scenario, the first server device *B* wishes to control the second sever device *A* in the network. The INTERFACE-A.XML document of the second device *A* is transferred from the second server device *A* to the first server device *B* and used by *Application B* to query the capabilities and API

35 interface methods of the second server device *A*. This allows the first server device *B* to control the second server device *A* utilizing XML remote procedure calls XMLRPC. In another scenario, the first server device *B* performs the above steps after attempting to communicate with the second server device *A* at least once, and failed to establish communication. Yet in

-28-

another scenario the first server device *B* queries the INTERFACE-A.XML document in the second server device *A* remotely without transferring the INTERFACE-A.XML document to the first server device *B*.

- 5           Upon examining the contents of the INTERFACE-A.XML document, the first server device *B* can create commands for sending to the second server device *A* in XML format as described above. Generally, the first server device *B* can interpret at least a portion of the contents of the INTERFACE-A.XML document that overlaps with a subset of the XCE definition used by
- 10   the first and second server devices *B* and *A* as described above. If the first server device *B* is unable to interpret a portion of the contents of the INTERFACE-A.XML document, then the first server device *B* can ignore that portion, or fetch an application to assist it in interpreting that portion, by translation as described further below.

- 15           Referring to Figure 17, another example device-to-device or inter-device control between a controller server device 14 and a controlled server device 14 is shown. The controller device 14 includes a controller application *E* and the controlled device 14 includes an application executable *C*. The
- 20   controlled device 14 further includes INTERFACE-A.XML, the application interface description *A* of the application *C*. Application *E* accesses the application interface description *A* in the controlled device 14 to query the capabilities and API interface methods of the controlled server device 14. Application *E* then commands and controls application *C* using XML remote
- 25   procedure calls to control hardware or service *D* of the controlled device 14. A scheduler device can be a case of a controller device 14, driven by time of day such as Time-Delay-Record controller in a VCR.

- 30           In a first example, the application *E* accesses the application interface description *A* by remote query over the network. In a second example, the application *E* accesses the application interface description *A* by transferring a copy of the application interface description *A* from the controlled device 14

-29-

to the controller device 14. The application *E* then queries the interface description *A* locally. In a third example, the application interface description *A* is transferred to a library device 64 which provides library space for interface descriptions, and the application *E* remotely queries the interface description *A* in the library. The library device 64 stored the address (URI) of the associated applications available for direct control action and responses.

Referring to Figure 18, the XML protocol provides a Web standard common middleware layer in a communication stack 68 at the API level between applications 20 of various devices 14 in the network. In each device 14, applications at the top of the communication stack send and receive communication messages over the network, and communicate with software layers in the device stack that locally control the device hardware or service software for the device.

A first XML layer API, designated as XML Layer OUT 68, is used for sending messages, and a second XML layer API, designated as XML Layer IN 70, is used for receiving messages. The XCE definition and the XML definition of a method call, namely the document type definition CALL.DTD described above, are used to create the XML Layer OUT 68. Further the XCE definition and the XML definition for a method call, namely document type definition INTERFACE.DTD described above, are used to create the XML Layer IN 70. For example a controller application utilizes the XML Layer OUT 68 and a controlled application utilizes the XML Layer IN 70.

Referring to Figure 19, another embodiment of server device-to-server device command and control architecture is shown. An XML-based control architecture is utilized for device-to-device (service to service) control for Web and Internet enabled devices or services. A first device *A* can remotely control an application 20 in a second device *B* over the network using XML command messages. The interface to each device includes interfaces to the applications in the device, and is described in XML format. Said interfaces

-30-

can be extended and made available on the middleware layer for retrieval and interpretation by other devices over the network, as described further below.

Each of the server devices *A* and *B* includes hardware and software for controlling other server devices over the network and for being controlled by other server devices over the network. In Figure 19, the home network device *A* is a controller device or module, and the home network device *B* is a controlled device or module. Each of the devices *A* and *B* includes a local Device XML Interface 72 comprising an interface document INTERFACE.XML and a document type definition INTERFACE.DTD. The INTERFACE.XML document includes a description of the objects, methods and parameters supported by the corresponding device 14. The INTERFACE.DTD document can be used for validity checks specific to the XML interface of the device, as described above.

Each of the devices *A* and *B* also includes an XML parser 74, comprising program code for parsing and validating XML messages, such as XML interface and XMLRPC commands. The XML parser 74 is similar to said XML Layer IN 70 described above with reference to Figure 18. Further, each of the devices *A* and *B* includes an XMLRPC encoder and decoder (codec) 76 for encoding method names and parameters of an outgoing call in an XMLRPC message, and for decoding an incoming XMLRPC message after it is parsed, to retrieve the method name and parameters therein. The XMLRPC codec 76 is independent of the device XML interface 72 and of the device-to-device control architecture, thereby allowing use of different XMLRPC formats without changing other aspect of the device to device control architecture.

An Interface Fetcher comprising program code, is utilized by each of the devices *A* and *B* to fetch the device interface of another device directly from another device or from a home network Interface Library 80. When a device 14 is a controller device, a controller application program code 82 in

-31-

the controller device 14 effects command and control of other devices 14 over the network, by supervising software and hardware in the controller device 14 such as the XML parser 74, the interface fetcher 78 and the XMLRPC codec 76. When a device 14 is a controlled device, a controlled application program code 84 in the controlled device 14 supervises software and hardware in the device 14 for the device 14 to be controlled by other devices 14. A Home Network Device Web server 86 in each of the devices *A* and *B* manages communication between the devices over the network. An XML to Native Lookup Table 88 in each of the devices *A* and *B* is used by the controlled application 84 to convert information in XMLRPC messages (e.g., method name, parameters name and type) to native interface of the device (e.g., native method name, parameters name and type). Said table 88 is not used when the names of methods and parameters in XML messages and the native interface of the device are the same.

Each of device the devices *A* and *B* further includes one or more Handlers 90, wherein each Handler 90 includes a pointer from within the controlled application 84 to a native implementation of one specific device functionality. In most devices, native implementations of device functionality include binary code at run-time. The binary code can be generated from higher level languages at compile time, including C and Java, for example. As such, consumer electronics manufacturers can add more Handlers 90 for new functions without affecting existing Handlers and function implementations. A hardware service 92 in each of the devices *A* and *B* includes native implementations of device functions. Each of the devices *A* and *B* also includes a Native Interface 94 which comprises the API of native implementation of the device functions.

Further, a Network Object Request Broker such as a Home Network Object Request Broker (HNORB) 79 and Interface Library (IL) 80 provides a middleware layer 98 for the home network 10. As shown in Figure 19, the middleware layer 98 can be located in a third device 96 or in a separate

-32-

- control hub. The HNORB 79 includes a software agent for use by one device 14 to discover the existence of other devices 14 connected to the network 10. The HNORB software agent organizes device names into a naming hierarchical tree structure, organizes device interfaces into said searchable
- 5 Interface Library, and provides device interfaces to a device requesting interface information.

- The middleware layer, comprising the HNORB 79 and the IL 80, can be connected directly to the Internet, such that selected home devices can be
- 10 accessed from outside of a local home network 10. The middleware layer 98 in one local home network can be connected to the middleware layer 98 in other local home networks over the Internet to provide an integrated network comprising two home networks 10. In that case, authorized users with the appropriate stream encryption can access a DVD changer in the user's
- 15 primary home, from a TV in the user's secondary home to play a video and view it on the TV.

- To use the Interface Library 80, at least one HNORB&IL should be running on the local home network 10. More than one HNORB&IL may also
- 20 be available. For example, a cable modem, several DTVs, and a central home hub can all have their own HNORB&IL software agents. To locate the HNORB&IL, a device 14 sends a broadcast message over the local home network. The first HNORB&IL to respond to the device 14 is utilized by the device 14. Once a HNORB&IL is located, the device 14 and the HNORB&IL
- 25 can establish a point-to-point Transmission Control Protocol (TCP) or User Datagram Protocol (UDP) connection for registration, interface request and fetch, and device lookup services. If a UDP protocol is not available, a TCP protocol can be used for high bandwidth connections such as IEEE 1394. HTTP-based XMLRPC can also be utilized for device to HNORB&IL
- 30 communications. For example, a device 14 can remotely call a "register" method of HNORB to pass the device interface as one or more parameters, or, a XMLRPC call can retrieve a partial or entire device interface from the IL

-33-

as a XMLRPC response or return value.

As aforementioned, more than one HNORB&ILs can run in a local home network 10 simultaneously, wherein each HNORB&IL recognizes a subset of available devices and one HNORB&IL can communicate with other HNORB&ILs to locate the devices 14 it can not find. Multiple HNORB&ILs on one local home network 10 can locate each other automatically by using broadcasting messages, such as UDP or TCP. In this case, multiple HNORBs construct a distributed object request broker, while multiple Interface Libraries 80 construct a distributed interface library. To provide fault tolerance, if one of the HNORB&IL should terminate unexpectedly, all devices registered with this HNORB&IL are notified and said devices can automatically register with another available HNORB&IL.

Each device interface has an associated consistent, unique logical name. Other devices can use said consistent, unique, logical name to recognize and access a device, even after said device's location or real network address has changed. The mapping of the logical names and real device addresses are handled by a software agent for naming service in HNORB. Preferably, a standardized naming method is utilized. More preferably, a hierarchical naming structure is used to organize device names into a hierarchical tree. This hierarchical structure can be expressed using "/", similar to that in a file system. The structure can be generated by different methods, such as by different service types as a home/MPEG2/TV; or by different locations, such as home/livingroom/VCR. Several naming trees may coexist for performance and efficiency.

In the example command and control between the controller server device A and the controller server device B in Figure 19, the middleware layer 98 is in the third device 96 or can be in a separate central hub. The grayed blocks show the device elements used for the specific command and control process depicted in Figure 19. In an example operation scenario, after the

-34-

devices *A* and *B* become available and accessible over the network, each device registers/submits itself and its XML interface to the central HNORB and IL middleware layer 98. If a central HNORB and IL middleware layer is not available, then each device broadcasts a message over the local home network to announce itself.

The controller application 82 of the device *A* attempts to query all or part of the device interface of the controlled device *B*. If an Interface Library 80 is not available, the controller device *A* can request and fetch the device interface of the controlled device *B* directly from the controller device *B* by first sending a request to device *B* over the network, and then receiving the XML interface of device *B* from the device *B*. However, if an Interface Library 80 is available, the controller device *A* can request all or part of the device interface of the controlled device *B* from the Interface Library 80. The software agent of HNORB obtains the XML device interface of the device *B* from the Interface Library 80 structure and sends it back to the controller device *A*.

Once the controller device *A* receives the XML device interface of the controlled device *B*, the controller application of device *A* uses the XML parser 74 of device *A* to parse and interpret the device interface of the device *B*. The XMLRPC codec 76 of device *A* then generates desired XMLRPC command messages using the parser results. The XMLRPC command messages are sent to the controlled device *B* over the network. Upon receiving said XMLRPC command messages, the controlled application 84 of device *B* uses the XML parser 74 of device *B* to parse and interpret the received XML command messages. The XMLRPC codec 76 of device *B* then decodes the parser results to obtain the method call information in the command message, including a method name and parameters for the device *B* functions to perform requested services.

The controlled application 84 of device *B* then uses the XML to Native Lookup Table 88 and Handlers 90 in the device *B* to access and launch the

-35-

native function implementations of device *B* through the native interface of device *B*. If a function generates any responses or return values, said responses or return values are encoded into XML or XMLRPC messages and sent to the controller device *A*. Further, the middleware layer HNORB and IL  
 5 can provide the controller device *A* with a reference to the controlled device *B*, whereby the device *A* can generate remote calls to the device *B* native functions just as calls to the local device *A* native function.

Preferably, a standard XMLRPC format is utilized so that all devices  
 10 can interpret and decode RPC calls over the network. Because the device interface of a controlled device 14 can be queried and examined by a controller device 14, preferably a simplified XMLRPC format with sufficient device interface information is utilized to improve efficiency. The following example shows two possible formats of XMLRPC calls for One Touch Record  
 15 (OTR) and Time Delayed Record (TDR) operations.

**EXAMPLE I:**

XML RPC call, example format including detailed tag and interface information:

- 20 1. Example of OTR call:  
 <?xml version="1.0"?>  
 <call>  
     <object>DVCR1.record</object>  
     <method>timeDelayedRecod</method>  
     <parameters>  
         <parameter>  
             <name>channel</name>  
             <value><int>4</int></value>  
         </parameter>  
         <parameter>  
             <name>recordTime</name>  
             <value><time>2:10:30</time></value>  
         </parameter>  
     </parameters>  
 </call>  
 35
2. Example of TDR call:  
 <?xml version="1.0"?>  
 <call>  
     <object>DVCR1.record</object>  
 40

-36-

```

    <method>oneTouchRecod</method>
    <parameters>
      <parameter>
        <name>channel</name>
        <value><channelName>NBC</channelName></value>
      </parameter>
      <parameter>
        <name>startTime</name>
        <value><datetime.iso8601>19990401T19:05:35</datetime.iso8601></value>
      </parameter>
      <parameter>
        <name>recordTime</name>
        <value><time>2:00:00</time></value>
      </parameter>
    </parameters>
  </call>

```

20 **EXAMPLE II:**  
XML RPC call, example format with reduced tags and interface information:

1. Example of OTR call:

```

  <?xml version="1.0"?>
  <call>
    <object>DVCR1.record</object>
    <method>timeDelayedRecod</method>
    <parameter value="4">channel</parameter>
    <parameter value="2:10:30">recordTime</parameter>
  </call>

```

2. Example of TDR call:

```

  <?xml version="1.0"?>
  <call>
    <object>DVCR1.record</object>
    <method>oneTouchRecod</method>
    <parameter value="NBC">channel</parameter>
    <parameter value="19990401T19:05:35">startTime</parameter>
    <parameter value="2:00:00">recordTime</parameter>
  </call>

```

Referring to Figure 20, device interfaces for home devices 14 are based on an industry standard structured data base 100 using standardized vocabulary. Interface data for new interfaces and vocabulary can be added to the data base 100. A comprehensive definition or database of CE objects, methods and parameters using English words to describe all CE devices is

-37-

termed a CE data base 102. The comprehensive definition or database can be in C, XML or other formats capable of representing objects and their respective methods and parameters. The comprehensive definition or database utilizing standardized XML vocabularies is termed XCE definition or data base 104.

Controller and controlled applications 82, 84 are programmed using a standard interface subset of the XML based XCE data base 104. Each device interface is stored with said applications 82, 84 in XML form. Although the XCE data base 104 need not be in XML, said subset interface produced at compile time is in XML in an embodiment of the invention, as described above in reference to Figure 15.

In Figure 20, for embedded appliances 14, the information designated as 'Manufacturer' information is built-in to the appliances 14 at manufacture time, and the information designated as 'Home Network' is part of the operational run time aspects of the appliance in the network. Device XML interfaces 72 designated as 1 ... N for N devices 14, are branches of the data in a standardized XCE data base 104. A Home Network Interface Library (HNIL) 106 provides a collection of the device interfaces of available devices 14 connected to the home network. The Home Network Interface Library 106 is a subset of the totality of the XCE data base 104.

In Figure 16, a device interface was transferred from a device A to a device B for an *Application B* in device B to examine the contents of the interface for the device A. As detailed above, a device interface includes a description of the objects, methods, parameters supported by a device, and is referred to as INTERFACE-A.XML for a device A for example. A Device XML interface 72 is a device interface in XML format. The content of the XCE data base 104 is a service oriented structure which provides device interfaces.

-38-

Referring to Figure 20, the XCE database 104 also includes a standardized XCE Interface Document Type Definition (DTD) for CE devices, which provides a standardized set of rules for using XML to represent CE devices 14. The DTD or its subsets can be used for validity checks. A software agent designated as Manufacturer Tool 108, filters and utilizes a subset of the standardized XCE definition 104 for a specific CE device, and uses the standardized XCE Interface DTD to generate an XML device interface 72 of the CE device, for example INTERFACE.XML and INTERFACE.DTD. The document INTERFACE.XML includes a description of the objects, methods and parameters supported by a specific device according to the standardized XCE Interface DTD. The document INTERFACE.DTD is a subset of the standardized XCE Interface DTD, and can be used for validity check for the XML interface of the device. Other document type definitions can also be used to create the INTERFACE.XML document.

The XML interfaces 72 of the CE devices, including said XML interface document and said DTD document, are stored in a universally accessible library such as the home network Interface Library 106. A software agent 110 collects the device interfaces 72 of all accessible devices 14 over the network and places them into the searchable structured Interface Library 106 along with the device name/address information. The Interface Library 106 is a subset of the XCE database 104 and the process of generating the Interface Library 106 is similar to that of rebuilding part or all of the XCE database 104. The Interface Library 106 can function as a collection of device interfaces 72 of all devices 14 in the home network, or as a cache depending on availability of storage space, wherein only the most recently used device interfaces 72 are stored therein. In cases where a device 14 updates its device interface 72 due to an event, such as disk change in a DVD player, part of the device interface 72 is updated based on an event service.

Referring to Figure 21, preferably the device interface definition 72 of

-39-

each device 14 has a hierarchical form. This is because for a home device 14, the device interface definition 72 can be lengthy. Typically, one or few functions such as a single function for Time Delayed Recording, are accessed at a time, and therefore only a small portion of the device interface 72 is used. Rather than rendering the entire device interface 72, it is more efficient to render only a portion of the device interface 72. By using hierarchical device XML interface, a controller device 14 can request partial device interface 72 of a controlled device 14 by specifying the desired function categories or functions in a request for the XML device interface from the controlled device 14 or from the HNORB and IL middleware layer 98. In the latter case, the HNORB and IL middleware layer 98 sends back the desired portion of the device interface 72.

Referring to Figure 21, the hierarchical device interface structure can include four layers, including: (1) a first layer 112 for XML interface of each home network, listing current available devices, (2) a second layer 114 for general XML interfaces of each device, listing function categories, (3) a third layer 116 for specific XML interface of each function category for a device, and (4) a fourth layer 118 for specific XML interface of each function in a function category. Inside the home network, only the three lower layers 114, 116 and 118 are utilized, and outside the home network the first layer 112 is utilized.

Figure 22 shows said layers 112, 114, 116, 118 and corresponding interface examples. The interface in each layer is linked to upper or lower layer (if available) through links such as XLink or XPointer, which provide two-way linking. XLink includes a package of hyperlinking functionality that has two parts: (1) an XLink component which allows links in an XML documents to be recognized as such, and (2) an XPointer component which allows links to address into precise sub-parts of an XML document. As such, XLink governs how links are inserted into XML documents, wherein the link may point to data such as a GIF file. Further, XPointer governs a fragment identifier that

-40-

can go on a URL when linking to an XML document, from anywhere (e.g., from an HTML file).

In a typical command and control model for a server device 14 to control another server device 14 according to the present invention, a first device 14 attempts to query the device interface of a second device 14 at the second interface layer 114. After selecting function categories (FC), the first device 14 queries the interface layer 116 of a specific function category in the second device 14, such as Record Category. Further, the first device 14 can query the interface layer 118 of a specific function, such as OTR or TDR, to make calls to said functions. The hierarchical or tree structure makes finding of an interface function more efficient and saves network bandwidth. An example interface file structure and layers can be:

First layer 112 - HN1.xml  
 Second layer 114 - VCR1.xml  
 Third layer 116 - VCR1\_RecordCategory.xml  
 Fourth layer 118 - VCR1\_RecordCategory\_OTR.xml

Similarly, the home network Interface Library 106 is preferably hierarchical and can be structured in a variety of ways such as by different service type of devices or by different locations such as rooms. Said hierarchical structure is the interface of a local home network 10 to other home networks or the Internet.

An example hierarchical device interface definition 72 which can be implemented in XML syntax is shown below.

```
consumer (document_file, doc)
  +----document_file<server_home.dtd, server_auto.dtd>
  +----doc (services_home, server_auto, server_samsung_web_site,
30  avc_commands, cal_commands,,)
    +----services_home (xml_utility, client, server_av, lighting,
      comms, hvac, utility, security, appliances, convenience,,)
      +----xml_utility (download_DTD_file,,)
35  +----client (acknowledge, attention, error, post_message,
    sound, stop_schedule, stop_all,,)
```

-41-

```

+----sound (alarm, ring, buzz,,)
+----server_av (controls_gen, source, sink)
+----controls_gen (ping, process_infor, setup,,)
+----process_info (s/w_id, h/w_id)
+----h/w_id (ser_no, manuf, model, class,,)
+----s/w_id (ser_no, exe_name, version,,)
+----setup (clock,,)
+----clock (hours, minutes, seconds)
+----source (service_id, media, rate, protocol,
10 stream_format, controls_av,,)
+----sink (service_id, media, rate, protocol, stream_format,
controls_av,,)
+----service_id (url,,)
+----media (tpt_stream, ram, disk, tape,,)
+----disk (name, number,,)
15 +----rate<value>
+----protocol (61883/1394, UDP/IP/Ethernet,,)
+----61883/1394 (isoch_ch_no)
+----stream_format (video, audio,,)
20 +----video (dv, mpeg2tpt, dsstpt, mpeg2pes,
mpeg10801-tpt,)
+----audio (mpeg3, ac-3, midi,,)
+controls_av (flow_control, tune, timer_record,
ui_control,,)
25 +----timer_record (tune, flow_control)
+----flow_control (play, stop, goto, record,,)
+----play (time_params)
+----record (time_params)
+----time_params (now, start, duration, end,,)
30 +----tune (send_epg, channel,,)
+----channel (number, id, time_params,,)
+----ui_control (display, acoustic)
+----display (brightness, contrast, color/tint,
horiz_size, vert_size,,)
35 +----acoustic (volumn, base, treble, balance, fade,)
+----lighting (sensors, lights, send_epg)
+----sensors (living_room, sky,,)
+----lights (rooms_up, rooms_down, yard,,)
+----rooms_up (bed1, bed2, bed3, bed4,,)
40 +----bed1 (lamp, dimmer,,)
+----dimmer<value>
+----rooms_down (family, kitchen, living, dining, soho,
garage,,)
+----yard (front, back)
45 +----comms (homehub, intercom, telco,)
+----homehub (send_device_list, send_configuration,
send_snmp_mib,,)
+----intercom ()
+----telco ()
50 +----hvac (controls_gen, controls_hvac,,)
+----controls_hvac (a/c, heat, temp, humidity,)
+----temp (low, high, hysteresis,,)
+----utility (meters, energy_mgmt,,)
+----meters (water, gas, electric,,)
55 +----water<value>, gas<value>, electric<value>

```

-42-

```

+----security (sensors, send_epg, alarm,,)
+----sensors (peripheral, motion,,)
+----peripheral (rooms_up, rooms_down,,)
+----motion (room_down, yard,,)
5 +----appliances (microwave, range, oven, fridge, freezer,
coffee, toaster, washer, dryer, water_heater,,)
+----microwave (send_epg, controls,,)
+----fridge (temp,,)
+----water_heater (temp)
10 +----convenience (window, curtain_open, door/gate, pool/spa,
bath, fountain, lift, jacuzzi,,)
+----curtain_open<value>
+----server_auto (message, server_auto_ford_explorer_98,,)
+----server_auto_ford_explorer_98 (mileage, maintenance,,)
15 +----mileage <data>
+----maintenance <data>
+----server_samsung_web_site (message, service, help,,,)
+----avc_commands<,,,command_string,,,>
+----service_id url,,)
20 +cal_commands<,,,command_string,,,>
+----service_id (url,,)

```

Said hierarchical device interface definition 72 can include the following fields:

'document file' name, provides name of the document type definition (DTD) file that can be used by an XML parser 74 for verification of legality and correctness of the XCE database 104 or part of the XML version of the XCE database 104. There can be several DTD files for different parts of the XCE structure, wherein said DTDs are different from the document type definitions for the RPC.CALL and INTERFACE.DTD for communication.

'doc' name, provides the top level name of the area of coverage of capabilities, attributes, communication and control interface.

'Services\_home', provides area for home automation, consumer electronics, utility, etc.

'Server\_auto', for an automobile in the garage and shows message interface available for one or more automobile types. For example, server\_auto\_ford\_explorer\_98' is the interface for a particular automobile. This allows access to mileage and maintenance interfaces of the automobile, and can also be used for remote access by an automobile manufacturer or garage for direct checking and remote diagnostics, for example.

'server\_samsung\_web\_site', provides for communicating with a manufacturer Web site outside the home. Includes interface for message,

service, help, etc.

'AVC\_commands' and 'CAL\_commands', provides for legacy devices capable of interpreting AV/C and CAL languages, for example. This portion of the structure identifies commands in said languages, where the commands are tagged and carried in XML. As such, the contents are not XCE (Web) objects, and protocol converter applications can be utilized to interface to the original CAL or AV/C application software.

In the above description, 'Services\_home' provides the main structure including A/V consumer electronics. A branch of the structure is expanded in detail for a particular example of a video services sink, and stream destination (e.g., DVCR) control interface. The control interfaces in a typical home network can include:

'xml\_utility', provides details for supporting utility network functions such as downloading an updated DTD file, interface file, program file, etc.

'client', describes the interface details of a client device including a Web browser. For example 'acknowledgment' indicates the controller's acceptance of acknowledgment of a message or command sent out.

'server\_av', provides control and capacity interfaces for all audio and video services available, including STB, DVCR, DTV, DVD, AUDIO, etc.

'lighting', provides an interface to a home automation lighting controller, and includes sensors, lights, etc.

'comms', provides control interfaces to communications devices, typically for utility purposes or remote management of the devices' set-up or parameters, or for restoring configurations.

'hvac', provides interfaces for remote control of the HVAC system, and can be used for control of said system from outside the home by the utility company, for example, to turn the home's HVAC system off during peak load periods of the day. Further, said interface can be used for controlling the HVAC system from within the home, by an appliance for device

-44-

based controller to provide a more sophisticated control mechanism than thermostat control.

'utility', provides interface for reading utility meters for the home, for example.

- 5           'security', provides interface for security sensors and alarm settings. As such, using the interface, applications running on a home network device can have access to the sensor and detector devices around the home for monitoring and controlling of the those devices.

- 'appliances', provides interfaces for kitchen, utility and general  
10   home appliances, including, for example, providing remote control or monitoring temperature settings or other controls and parameters from a controller device. In one scenario, a microwave appliance can scan bar code information on the packaging of a food item and access a manufacturer database to obtain cooking time of the food for a given microwave system  
15   type. Such integration of appliances using device to device command and control provides many control scenarios for providing services such as automatically pausing a dishwasher and muting a TV when a phone is picked up in the kitchen or family room.

- 'convenience', provides interfaces to devices for providing  
20   convenience services such as interface to a curtain, window, blinds or whirlpool controllers, for example.

In the above description, 'server\_av' is part of the structure for the control interfaces for A/V appliances offering A/V stream service, and is subdivided into 'controls-gen', 'source' and 'sink' capabilities.

- 25           'controls-gen', provides interface for device manufacturer attributes and general utility interfacing such as ping testing the presence of the device. Further, manufactured-in attributes such as software and hardware identification and version information can also be included. A device supplying this interface returns data providing name or identification  
30   for said software without effecting any control actions. An interface to set the time of day clock can also be included.

          'sink', provides interface for the media stream service devices.

-45-

The structure is organized based on service offered (i.e. video stream record and replay) rather than particular device names such as VCR. For example, a Tuner and a DVD player are both video program stream sources for the network with video program formats, and can be controlled, such as started and stopped. Differences in control of particular devices are addressed by the lower layers of the definition structure.

'source' provides interface similar to the 'sink' interface.

Referenced above, 'service\_id' or 'application\_interface\_id' includes the name, address or Web address or URL location of one or more devices 14. Because the XCE database 104 comprises the totality of agreed upon interfaces, typically a Dynamic Host Configuration Protocol (DHCP) software agent executes to assign an address and a default name to each device, and the address and a default name are added the interface of the service or device. The software agent 110 then collects device interfaces 72 which include subset or 'device partial XCE' definitions from all the devices locally connected to the home network to generate a 'network partial XCE'. Additional relevant external interfaces can be added to the structure for external control. For example 'service\_id' can be a name/address in a received structure or in a network Interface Library 106 including entries from the software agent according to the device interfaces of the devices connected to the network. Thereafter, a user can search for a service in the database and access an application whose interface includes a particular data branch of the library using said name/address. As such, the network can include multiple identical services distinguished by said name/address information.

'media', provides interface for the type of media including, for example, transport stream from a tuner, RAM from a PC DRAM, disk for CD or DVD, and tape. The media can be named and identified, and a controller device can search the XCE data base to identify the media currently provided on the network. When a new media such as DVD disk is provided on the network,

-46-

that portion of the device interface 72 identifying the program material on the disk is changed accordingly. As such, the entire device interface 72 need not be transferred and only the relevant portion is transmitted to the XCE data base. On receipt of an attention signal, the library software agent 110 can  
5 fetch the new update and place it in the proper place in the interface library 106. The addition of the disk media is similar to adding a service to the network or connecting another appliance to the network.

'rate', provides a value for data stream rate for a device interface, such as 6 Mbits/Sec or 19.2 Mbits/Sec, for example.

10 'protocol', identifies the protocol used for said data stream. If more than one protocol is provided, for example 61883/1394 or UDP/IP, then a desired protocol can be selected.

'stream\_format', provides packet format and/or compression standard for digital stream audio and video split. If more than one format is provided, a  
15 desired format can be selected via an interface message. A controller application 82 can examine the available formats to determine if there are compatible ones.

'controls\_av', provides the main control interface for A/V media appliance.

20 'Flow\_control', provides data stream controls such as: PLAY, STOP, GOTO, RECORD, etc as methods for a particular device. The methods do not change for embedded appliance, except for PC software, for example. The controls can include time parameters for delayed operation.

'Tuning', provides interface for tuning control. A controller device 14  
25 can send a request to the interfaces of a controlled device 14 to send back an Electronic Program Guide (EPG) data structure described above.

'UI control', provides control interface to a controlled application 84 to control adjustments for display such as brightness and contrast, and for audio, such as volume and bass.

30 'Timer\_record' provides interface for set-up data for a controller application 82 to implement delayed time recording. Direct channel tune information and flow control (time\_aparams) information can be utilized.

-47-

The above description can apply equally to client devices 12. An alternative syntax XCE definition or database for the CE space can be utilized. The alternative syntax XCE data base includes full service descriptions including home automation, appliances and automobile, for example. In cases where a service object provides flexibility and parameters for control, a control method is utilized to control the object as desired. Example commands in the AV/C and CAL command languages including binary and hex data strings are shown below.

```

10 consumer (document_file, doc)
    +----document_file<server_home.dtd, server_auto.dtd>
    +----doc {avc_commands, cal_commands, services_home,
server_auto,
15 server_samsung_web_site, server_auto_ford_explorer_98,,)
    +----avc_commands<...command_string...>
    +----cal_commands<...command_string...>
    +----services_home (client, av, lighting, comms, hvac,
utility, security,
20 appliance, convenience,,)
    +----xml_utility (download_DTD_files,,)
    +----client (acknowledge, attention, error,
post_message, sound,
stop_schedule, stop_all,,)
    +----sound (alarm, ring, buzz,,)
25 +----server_av (source, sink)
    +----source (service_id, media, rate, protocol,
stream_format,
controls_gen, controls_av,,)
    +----sink (service_id, media, rate, protocol,
30 stream_format,
controls,,)
    +----service_id (url,,)
    +----media (tpt_stream, ram, disk, tape,,)
    +----disk (name, number,,)
35 +----rate<value>
    +----protocol (61883/1394,
UDP/IP/Ethernet,,)
    +----61883/1394 (isoch_ch_no)
40 +----stream_format (video, audio,,)
    +----video (dv, mpeg2tpt, dsstpt,
mpeg2pes,
mpeg1080i-tpt,)
    +----audio (mpeg3, ac-3, midi,,)
    +----controls_gen (ping, process_info,
45 setup,,)
    +----controls_av (flow_control, tune,
timer_record,
ui_control,,)
    +----process_info (s/w_id, h/w_id)

```

-48-

```

class,,)
version,,)
5 seconds)
record,,)
10 time_params,,)
start, duration, end,,)
contrast,
color/tint, horiz_size, vert_size,,)
treble,
balance, fade,)
25 soho,
garage,,)
yard (front, back)
bed1 (lamp, dimmer,,)
dimmer<value>
35 comms (netman, intercom, telco,)
netman (send_device_list,
send_configuration,
send_snmp_mib,,)
40 intercom ()
telco ()
hvac (controls_gen, controls_hvac,,)
controls_hvac (a/c, heat, temp, humidity,)
temp (low, high, hysteresis,,)
45 utility (meters, energy_mgmt,,)
meters (water, gas, electric,,)
water<value>, gas<value>,
electric<value>
security (sensors, send_epg, alarm,,)
50 sensors (peripheral, motion,,)
peripheral (rooms_up, rooms_down,,)
motion (rooms_down, yard,,)
appliances (microwave, range, over, fridge,
freezer, coffee, toaster,
washer, dryer, water-heater,,)
55 microwave (send_epg, controls,,)

```

-49-

```

                    +----fridge (temp,,)
                    +----water-heater (temp)
                    +----convenience (window, curtain_open, door/gate,
5 pool/spa, bath,
   fountain, lift,,)
                    +----curtain_open <value>
                    +----server_auto (message, mileage, maintenance,,)
                    +----mileage<date>
                    +----maintenace<data>
10

```

In another aspect, the present invention provides for use of existing command language implementations for device-to-device command and control in a network. Devices can include internal objects and APIs which, at run time, create binary strings according to existing transport mechanisms. In that case, in order to provide XML remote procedure calls (XML RPC) from one device 14 to another device 14 in the network, the exiting application interface implementation is replaced with calls to the XML service API. As such, the original implementation is equivalent to a wrapper for the XML service API. Figure 18 also shows applications created using other command languages such as CAL or AV/C in dashed lines, with their interface implementations replaced with a wrapper in the XCE/XML service API. Examples for changing from CAL command language to XML RPC format are shown below.

```

25 -----
   existing implementation:

   void DeviceCALCommand (int command) {
30       .
       /*
       create CAL formatted byte string to represent this
       object/method and output to the wire
       */
35       CreatCALFormattedByteString(command); /* different for
                                                every protocol */
       SendCALByteString(); /* different for every protocol
40   */
   }

   -----
   wrapping the XML Service API call:

   void DeviceCALCommand (int command) {

```

-50-

```
{  
    /*  
    replace CAL implementation with calls to the XML  
ServiceAPI  
5    */  
    CreateXMLMessage (command); /* always the same */  
    sendXMLMessage ();          /* always the same */  
10 }
```

Referring to Figure 23, in another aspect, the present invention provides a standard command protocol and control language translation for inter-device communication between disparate devices in a network. For different devices to share information, the information must be in a format that  
15 a requesting device can interpret. And, for a device 120 to control another device 22, the two devices must use a common language in order to interpret one another's commands. The present invention provides a common identification format for data and command protocols.

In one embodiment, a method of common presentation or packaging of data and command protocol is provided, whereby a receiving device 122 can determine the native format of transmitted data. If the receiving device 122 can interpret that native format, then it can accept the data directly. Otherwise, the receiving device 122 can request a translator device 124 or  
25 application to translate the data into a desired format which the requesting device 122 can interpret. The translator device 124 or application determines the native format of the original data, translates the data into said desired format, and sends the translated data to the requesting device 122.

The requesting device 122 then processes that data as though the  
30 data had originally been provided in the requesting device's native language format by the sending device 120. The requesting device 122 can also send a response back to the sending device 120 in the requesting device's native format, or send a response by proxy through the translator device 124 or  
35 application for translation into the native format of the sending device 120. The translation method can be utilized for information including command

-51-

protocols, data files and audio/video streams.

For devices that do not utilize the common format described above, the present invention provides for translation of data including command protocols to, and from, such non-compliant devices. For example, when a non-compliant device 120 sends data to a compliant device 122, the compliant device 122 can translate the data based on a determination of the native format of the data. For example, the compliant device 122 can examine the data for particular bit patterns within the data. When a compliant device sends data to a known non-compliant device, the compliant device can translate the data before transmission based on a determination of the native format of the non-compliant device.

An example implementation can be for a home network which supports the IP and HTTP protocols. The home network can be connected to the Internet to obtain applications and services of various types for desired functionality. As such, the common format method can be made compatible with Internet protocols and procedure for operation over the Internet and the home network.

One example of providing a common data format is utilizing XML to create a package for the data for transmission over the home network. The data can include command protocol, streaming audio or video, graphics or applications. The data is 'wrapped' with a standard header identifying the native format of the data and contents of the package, in XML form. The header allows unique identification of the data type the data portion of the XML code, whereby the data can be translated if necessary and provided to appropriate applications upon receipt.

Under the Web standard, the identification process is performed by browsers using file name extensions to identify the type and contents of a file transmission. The browsers then launch appropriate plug-in modules to

-52-

process the file. In the home network, XML is utilized to identify data transmissions which provides all home network transmissions over IP with a common identification method as described above.

- 5           Alternatively, a software layer can be provided in the home network protocol stack to uniquely identify the contents of all data transmissions over the home network. The software layer can be used instead of XML. The common format and identification principles of the present invention apply equally in either embodiment using XML or said software layer as
- 10   identification methods.

- In Figure 23, upon receipt of a data package transmission, the receiving device 122 examines the XML identity header of the data package to determine the format of the data therein. If the data is in a format
- 15   recognizable by the device 122, the XML identity header information is discarded and the device processes the data directly. Otherwise, the device 122 converts the received XML package into an XML translation request package and sends the request package and the data to the translation server device 124.

- 20           The translation server device 124 translates the data and converts the translated data into an XML translation response package. The translation server 124 then transmits the response package back to the requesting device 122. In case of a translation error, the translation server 124 can
- 25   provide a translation response error condition to the requesting device 122. Upon receiving the translated data, the requesting device 122 processes the translated data in the response package.

- 30           Example of an XML data package or packet can be:  
          <IDENTITY type=format=AV/c>...packet data ...<IDENTITY>

          Example of a translation request package or packet can be:  
          <TRANSLATION REQUEST type=Command format=CAL>  
          <IDENTITY type=Command format=AV/C>...packet data

-53-

...</IDENTITY>  
 <\TRANSLATION REQUEST>

5       Example of a translation request package or packet can be:  
       <TRANSLATION RESPONSE type=Command  
 format=CAL>...packet           data ...  
       <\TRANSLATION RESPONSE>

10       Example of a translation response error condition package or packet  
       can be.  
       <TRANSLATION RESPONSE type=Command  
 format=CAL>...packet data ...  
       <ERROR condition=Unrecognized command>Translation could  
       not be                   performed<|ERROR>  
 15       <\TRANSLATION RESPONSE>

Further, Table 3 in Figure 24 includes a partial list of package or  
 packet types and formats.

20       To provide translation services, a translation server 124 is identified in  
 the network during network configuration in a manner similar to that of DHCP  
 servers. The translation server 124 broadcasts its IP address to all devices in  
 the network for a period of time after the network is configured. All devices  
 120, 122 compatible with the translation services store the IP address of the  
 25       translation server 124 as it is broadcast over the network during network boot  
 up.

Alternatively, the requesting device 122 can broadcast a translation  
 request over the home network. All translation servers 124 in the network  
 30       that receive the translation request can respond to the translation request by  
 sending a translation response to the requesting device 122. The requesting  
 device 122 then selects one translation server 124 among the responding  
 translation servers. In one example, the requesting device 122 selects the  
 first translation server 124 that responds to the translation request. In  
 35       another example, the translation servers 124 can negotiate among  
 themselves and/or with the requesting device 122 for the selection of a  
 translation server 124 for satisfying the translation request.

-54-

In another embodiment of the invention, multiple translation servers 124 are utilized to fulfill all translation requests. For example, a single translation server 124 may not have the capability to translate all requests. In such cases, it is necessary to identify the address of each translation server 124 and the type of translation service each translation server 124 can provide. Each device 120, 122 can store a list of all translation server IP addresses and a corresponding list of the types of translation services each translation server 124 provides, and optionally the associated translation application.

10

For efficiency, if a sending device 120 wishes to send data to a receiving device 122 which is known to use a different native format than that of the sending device 120, the sending device 120 can send the data to the receiving device 122 by proxy through a translation server 124. The sending device 120 transmits a command to the translation server 124 similar to the translation request command and includes the address of the receiving device 122 as the destination for the translated data.

In cases where a receiving device 122 requires translation of a data stream, the sending device 120 can route the data stream directly to a translation server 124, and the translation server 124 in turn transmits the translated data to the receiving device 122 as described above. Alternatively, the sending device 120 can send the data stream to the receiving device 122, and the receiving device 122 then routes the data stream to the translation server 124 for translation and return of the translated data back to the receiving device 122.

In the description herein, the control mechanism is based on the Hypertext Transfer Protocol (HTTP 1.1) which provides an application-level protocol for distributed, collaborative, hypermedia information systems. HTTP is a generic, stateless, object-oriented protocol in wide use for many tasks. A feature of HTTP is the typing and negotiation of data representation, allowing

-55-

systems to be built independently of the data being transferred. Preferably, the network protocol used by devices and applications on the home network is IP (Internet Protocol). However, other protocols can also be utilized.

- 5           Although the present invention has been described in considerable detail with regard to the preferred versions thereof, other versions are possible. Therefore, the appended claims should not be limited to the descriptions of the preferred versions contained herein.

**What is claimed is:**

1. A method for performing a service on a home network, the method comprising the steps of:
  - 5 (a) connecting a first home device to the home network;
  - (b) connecting a second home device to the home network;
  - (c) providing a database including a plurality of application interface description data objects, each application interface description data object including information in a structured format for commanding and  
10 controlling of a home device by one or more other home devices connected to the network;
  - (d) the second home device accessing a first application interface description object for the first home device in the database;
  - (e) the first home device accessing a second application  
15 interface description object for the second home device in the database;
  - (f) sending control and command data from the first home device to the second home device utilizing said application interface description object for the second home device over the network; and
  - (g) sending control and command data from the second  
20 home device to the first home device utilizing said application interface description object for the first home device over the network;whereby the first and second home devices perform said service.
2. The method of claim 1 wherein the structured format includes  
25 XML format.
3. The method of claim 1 wherein step (c) includes connecting a database device to the network, wherein the database device stores said database.  
30
4. The method of claim 3 wherein:
  - (i) the first home device stores the first application interface

-57-

data therein;

(ii) the second home device stores the second application interface data therein; and

- (iii) step (c) includes an initial step of forming said database  
5 by steps including querying the first and second home devices to transfer said application interface data for the first and second home devices to the database device.

5. The method of claim 1 wherein step (d) includes providing the  
10 first application interface description object for the first home device from the database to the second home device over the network.

6. The method of claim 1 wherein step (e) includes providing the  
15 second application interface description object for the second home device from the data base to the first home device over the network.

7. The method of claim 1 further comprising connecting three or  
more home devices to the network, wherein at least one home device  
20 accesses the database to query the application interface description objects of a plurality of home devices for sending command and control data to the plurality of home devices over the network.

8. The method of claim 1 wherein each application interface  
description object includes data in a structured format.

25

9. A network system for providing a service, comprising:

- (a) a physical layer, wherein the physical layer provides a  
communication  
medium than can be used by devices to communicate with each other;  
30 (b) first home device;  
(c) a second home device;  
(d) a database including a plurality of application interface

-58-

description data objects, each application interface description data object including information in a structured format for commanding and controlling of a home device by one or more other devices connected to the network; wherein:

- 5                   the second home device includes application control means for accessing a first application interface description object for the first home device in the database and sending control and command data from the second home device to the first home device utilizing said first application interface description object; and
- 10                  the first home device includes application control means for accessing a second application interface description object for the second home device in the database and sending control and command data from the first home device to the second home device utilizing said second application interface description object;
- 15                  whereby the first and second home devices perform said service.

10.   The network system of claim 9 wherein the structured format includes XML format.

- 20                  11.   The network system of claim 9 further comprising a data base device storing said database.

12.   The network system of claim 11 wherein:

- 25                  (i)    the first home device stores first application interface description object therein;
- (ii)   the second home device stores second application interface description object therein; and
- (iii)   said database base device forms said date base by querying the first and second home devices to transfer said first and second
- 30                  application interface description objects, respectively, to the database device.

13.   The network system of claim 9 wherein the control application

means of the second home device obtains the first application interface description object for the first home device from the database.

14. The network system of claim 9 wherein the control application  
5 means of the first home device obtains the second application interface description object for the second home device from the data base.

15. The network system of claim 9 further comprising three or more  
home devices, wherein at least one home device accesses the database to  
10 query the application interface description objects of a plurality of home devices for sending command and control data to the plurality of home devices over the network.

16. The network system of claim 9 wherein each application  
15 interface description object includes data in a structured format.

17. The network system of claim 9 wherein the structured format includes XML format.

FIG. 1

## Building Blocks or Modules

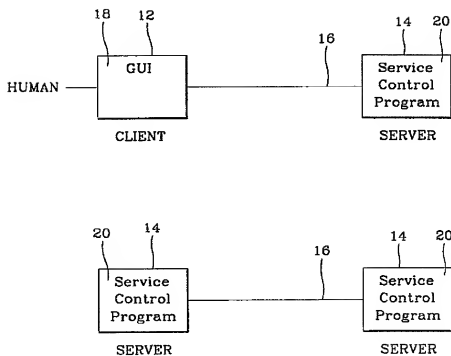


FIG. 2

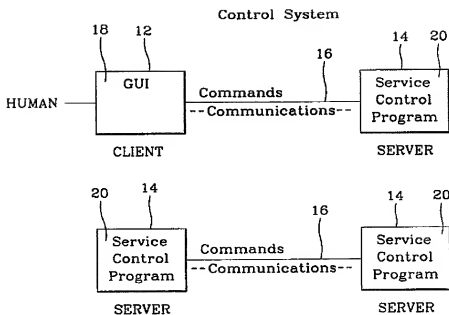


FIG. 3

Multiple Client Multiple Server Home Network Control Model

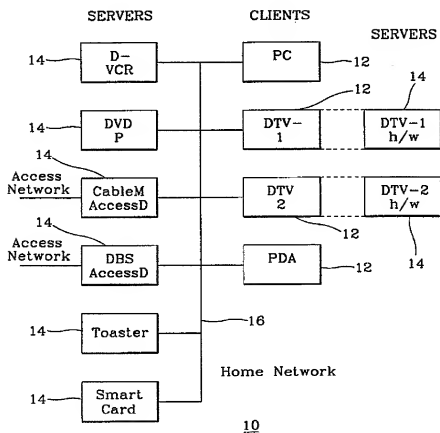


FIG. 4

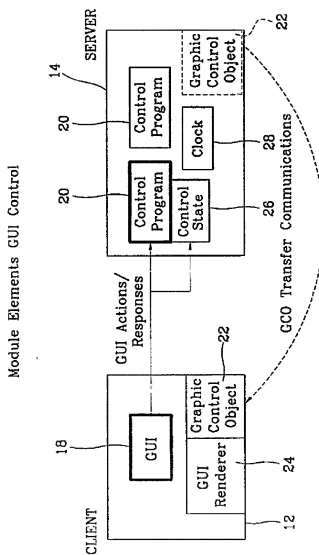


FIG. 5

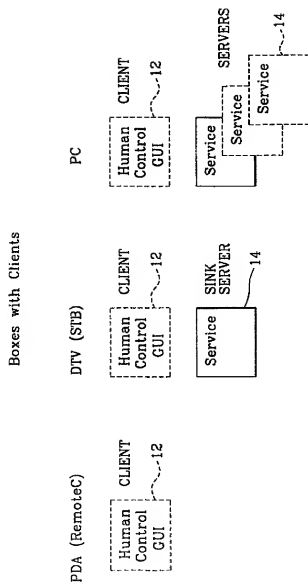


FIG. 6

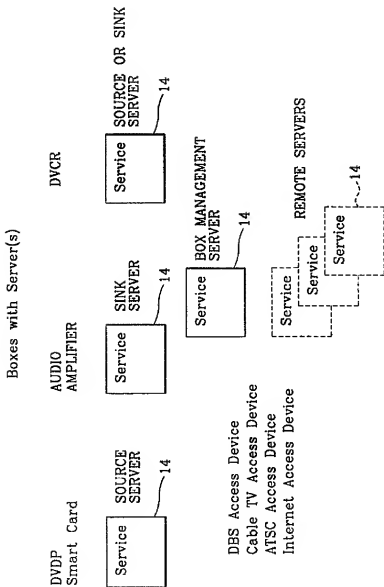


FIG. 7

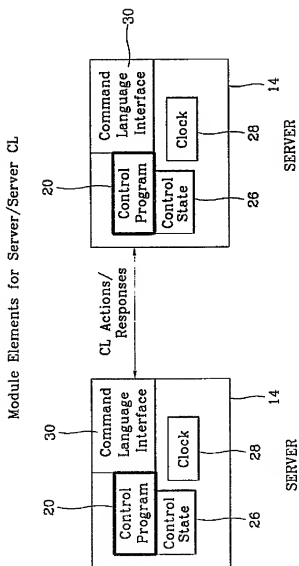


FIG. 8

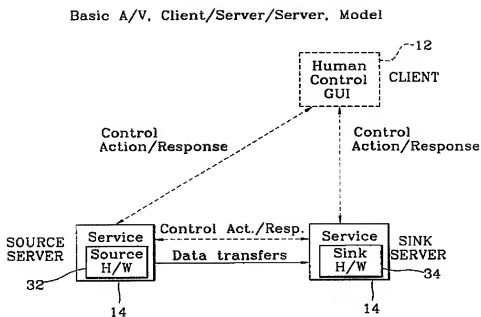


FIG. 9

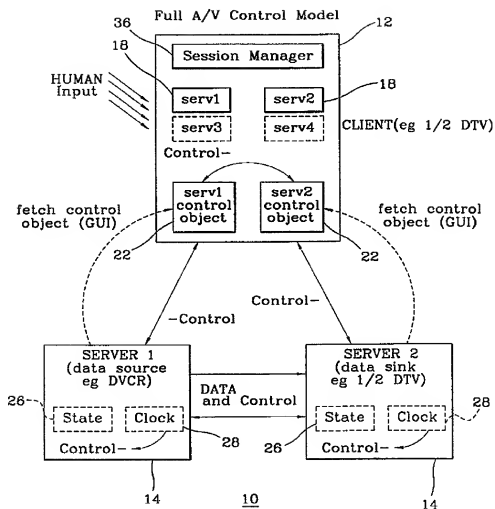


FIG. 10

Table 1: Capabilities Tavble

Service	Sink/Source
Service A	Sink
Service B	Source
Service C	Source
Service D	Sink

FIG. 11

Table 2: Attributes Table

NAME	LENGTH	VALUE
DeviceManufacturer	20 chars	Device manufacturer's name
ManufacturerURL	60 chars	Device manufacturer's home page URL
ManufacturerIcon	20 chars	Name of Device manufacturer's icon
DeviceName	30 chars	Device name
DeviceModel	20 chars	Device model number
DeviceType	20 chars	Device category
DeviceLocation	30 chars	Device group or location
DefaultSource	15 chars	Data type, Default source device (IP address)
DefaultSink	15 chars	Data type, Default sink device (IP address)
DeviceIcon	20 chars	Name of device icon

FIG. 12

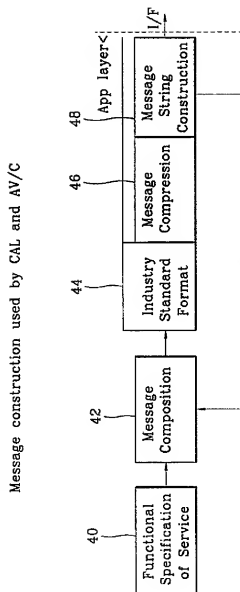


FIG. 13

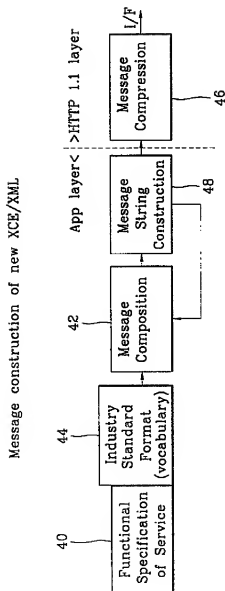


FIG. 14

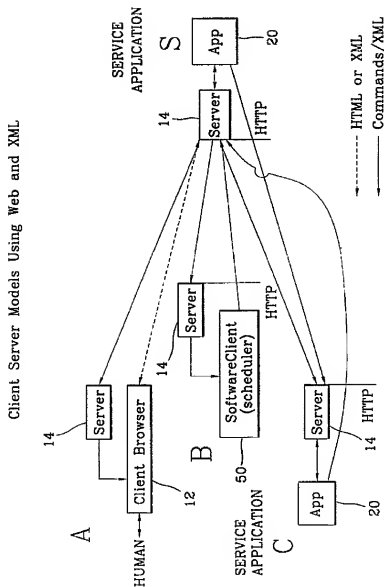


FIG. 15

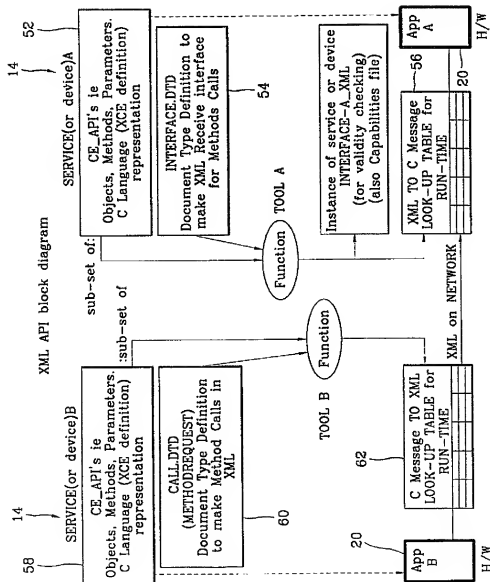


FIG. 16

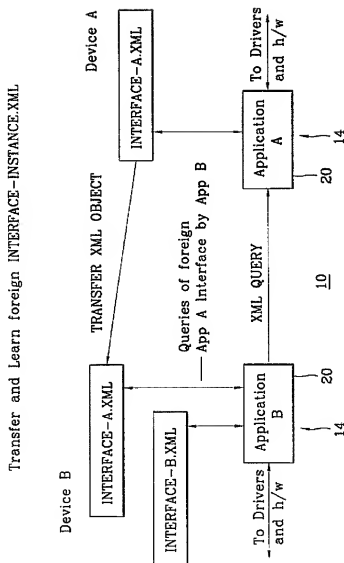
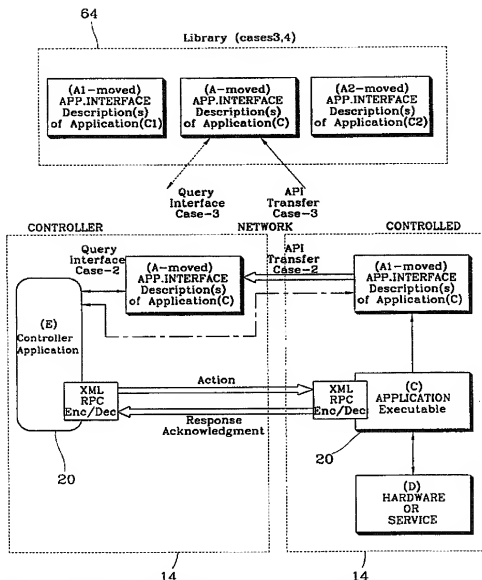


FIG. 17



Case 1. XML Application-C INTERFACE(A) for remote query by (E)  
 Case 2. Move XML Application-C INTERFACE(A) to controller for local query by (E)  
 Case 3. Move XML Application-C INTERFACE(A) to 3rd party device-a collecting place (library) for allinterface for remote query by (E). The library would have to have the address (URL) of the associated application available for direct control action and responses.  
 Case 4. Is case 3 but the 'indirect' control action (and response) is also directed at the library device. In this case the library is built with a 'forwarder'

FIG. 18

## XCE/XML Protocol Stack

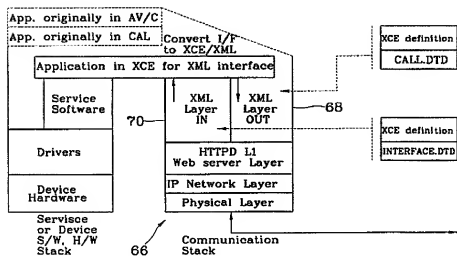


FIG. 19

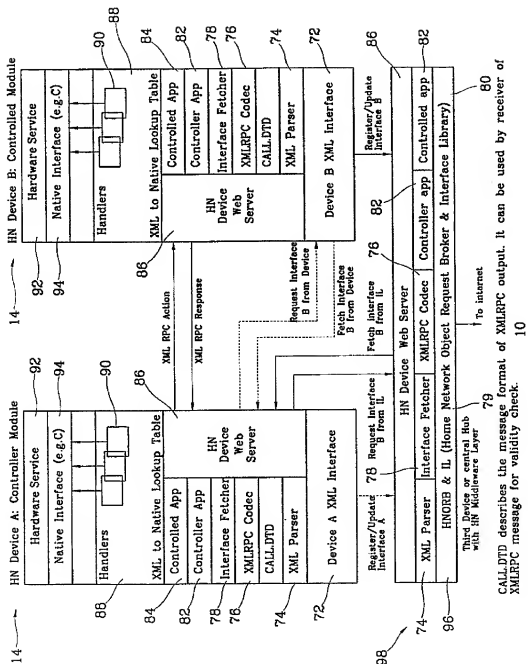


FIG. 20

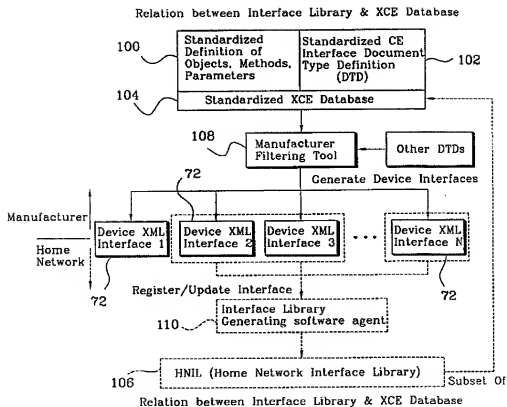
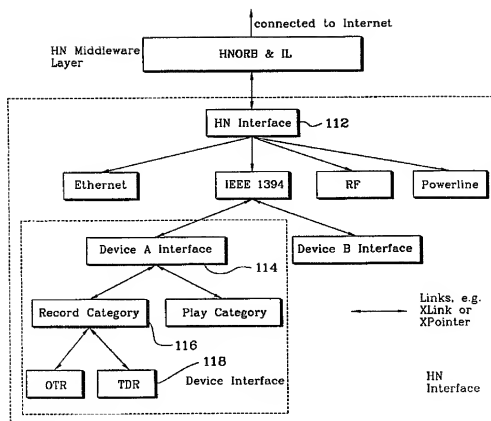


FIG. 21



Example of Hierarchical IL &amp; Device Interface

FIG. 22

Hierarchical Interface Structure. Four layers.

Layer	Interface Layer	Interface Example
112 1	XML interface for each HN, listing current available devices.	MyHN(VCRs, TVs...)
114 2	General XML interface for each device, listing function categories.	General VCR interface. (Record, play...)
116 3	Specific XML interface for each function category for a device.	VCR record category
118 4	Specific XML interface for each function in function category.	VCR record idr function

FIG. 23

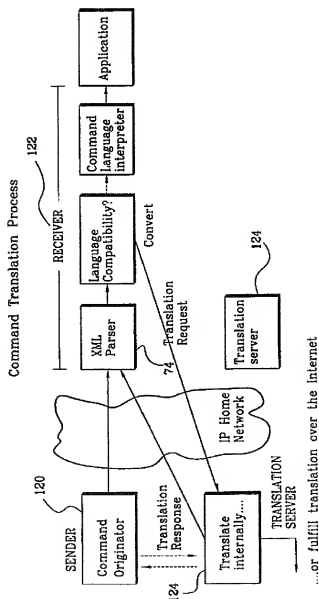


FIG. 24

Table 3: Example Formats and Types

Type	Format
command	CAL, AV/C, X-10
image	jpeg, gif, bitmap, tiff
language	English, French, etc
video_stream	mpeg2
video_clib	avi, quicktime, mpeg
text	html, plain
audio	wav, aiff
application	msword, pdf, postscript, gzip





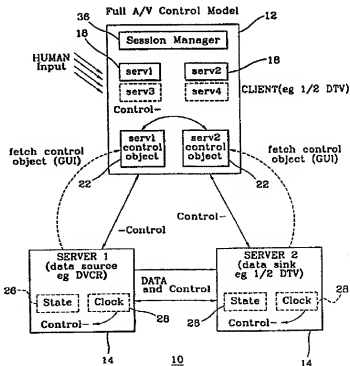
## INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 7 : <b>H04N 7/00</b>		A3	(11) International Publication Number: <b>WO 99/57837</b>
			(43) International Publication Date: 11 November 1999 (11.11.99)
(21) International Application Number: PCT/KR99/00221		(81) Designated States: AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, CA, CH, CN, CU, CZ, DE, DK, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MD, MG, MK, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, UA, UG, UZ, VN, YU, ZW, ARIPO patent (GH, GM, KE, LS, MW, SD, SL, SZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).	
(22) International Filing Date: 7 May 1999 (07.05.99)			
(30) Priority Data: 60/084,578 7 May 1998 (07.05.98) US			
(71) Applicant: SAMSUNG ELECTRONICS CO., LTD. [KR/KR]; 416, Maetan-dong, Paldal-gu, Suwon-city, Kyungki-do 442-373 (KR).			
(72) Inventors: HUMPLEMAN, Richard; 343, Lower Vintners Circle, Fremont, CA 94539 (US). WANG, Dongyan; 440 Oak Grove Drive #302, Santa Clara, CA 95054 (US).			
(74) Agent: LEE, Young, Pil; The Cheonghwa Building, 1571-18 Seocho-dong, Seocho-gu, Seoul 137-073 (KR).			
		<p><b>Published</b>  <i>With international search report.</i>  <i>Before the expiration of the time limit for amending the claims and to be republished in the event of the receipt of amendments.</i></p>	
		(88) Date of publication of the international search report: 29 June 2000 (29.06.00)	

(54) Title: METHOD AND APPARATUS FOR UNIVERSALLY ACCESSIBLE COMMAND AND CONTROL INFORMATION IN A NETWORK

## (57) Abstract

A method and system for performing a service on a home network, by: connecting a first and a second home device to the home network; providing a database including a plurality of application interface description data objects, where each application interface description data object includes information in a structured format for commanding and controlling of a home device by one or more other home device connected to the network; the second home device accessing a first application interface description object for the first home device in the database; the first home device accessing a second application interface description object for the second home device in the database; sending control and command data from the first home device to the second home device utilizing the second application interface description object over the network; and sending control and command data from the second home device to the first home device utilizing the first application interface description object over the network. Whereby, the first and second home devices perform said service.



**FOR THE PURPOSES OF INFORMATION ONLY**

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AL	Albania	ES	Spain	LS	Lesotho	SI	Slovenia
AM	Armenia	FI	Finland	LT	Lithuania	SK	Slovakia
AT	Austria	FR	France	LU	Luxembourg	SN	Senegal
AU	Australia	GA	Gabon	LV	Latvia	SZ	Swaziland
AZ	Azerbaijan	GB	United Kingdom	MC	Monaco	TD	Chad
BA	Bosnia and Herzegovina	GE	Georgia	MD	Republic of Moldova	TG	Togo
BB	Barbados	GH	Ghana	MG	Madagascar	TJ	Tajikistan
BE	Belgium	GN	Guinea	MK	The former Yugoslav Republic of Macedonia	TM	Turkmenistan
BF	Burkina Faso	GR	Greece	ML	Mali	TR	Turkey
BG	Bulgaria	HU	Hungary	MR	Mauritania	TT	Trinidad and Tobago
BJ	Benin	IE	Ireland	MN	Mongolia	UA	Ukraine
BR	Brazil	IL	Israel	MR	Mauritania	UG	Uganda
BY	Belarus	IS	Iceland	MW	Malawi	US	United States of America
CA	Canada	IT	Italy	MX	Mexico	UZ	Uzbekistan
CF	Central African Republic	JP	Japan	NE	Niger	VN	Viet Nam
CG	Congo	KE	Kenya	NL	Netherlands	YU	Yugoslavia
CH	Switzerland	KG	Kyrgyzstan	NO	Norway	ZW	Zimbabwe
CI	Côte d'Ivoire	KP	Democratic People's Republic of Korea	NZ	New Zealand		
CM	Cameroon	KR	Republic of Korea	PL	Poland		
CN	China	KZ	Kazakhstan	PT	Portugal		
CZ	Czech Republic	LC	Saint Lucia	RO	Romania		
DE	Germany	LI	Liechtenstein	RU	Russian Federation		
DK	Denmark	LK	Sri Lanka	SD	Sudan		
EE	Estonia	LR	Liberia	SE	Sweden		
				SG	Singapore		

# INTERNATIONAL SEARCH REPORT

International application No.  
PCT/KR 99/00221

## A. CLASSIFICATION OF SUBJECT MATTER

IPC<sup>7</sup>: H 04 N 7/00

According to International Patent Classification (IPC) or to both national classification and IPC

## B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

IPC<sup>7</sup>: H 04 N; H 04 L; G 06 F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

EPODOC

## C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	US 5546584 A (LUNDIN et al.) 13 August 1996 (13.08.96), abstract; fig. 1-4	1,9,12
A	WO 96/21189 A1 (COMPUSERVE) 11 July 1996 (11.07.96), claim 1; fig. 1A.	9
A	EP 0814403 A1 (H.P.) 29 December 1997 (29.12.97), abstract; fig. 1.	1
A	EP 0784271 A2 (NEC) 16 July 1997 (16.07.97), claim 3; fig. 1.	1

☐ Further documents are listed in the continuation of Box C.

☒ See patent family annex.

\* Special categories of cited documents:

„A“ document defining the general state of the art which is not considered to be of particular relevance

„E“ earlier application or patent but published on or after the international filing date

„I“ document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

„O“ document referring to an oral disclosure, use, exhibition or other means

„P“ document published prior to the international filing date but later than the priority date claimed

„T“ later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

„X“ document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

„Y“ document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

„&“ document member of the same patent family

Date of the actual completion of the international search

19 April 2000 (19.04.00)

Date of mailing of the international search report

03 May 2000 (03.05.00)

Name and mailing address of the ISA/AT

Austrian Patent Office  
Kohlmarkt 8-10; A-1014 Vienna  
Facsimile No. 1/53424/200

Authorized officer

Mihatsek

Telephone No. 1/53424/329

**INTERNATIONAL SEARCH REPORT**  
Information on patent family members

International application No.  
PCT/KR 99/00221

Patent document cited in search report			Publication date	Patent (family member(s))		Publication date	
US	A	5546584	13-08-1996	AU	A1	45168/93	31-01-1994
				AU	B2	686105	05-02-1998
				BR	A	9306694	08-12-1998
				CN	A	1080751	12-01-1994
				EP	A1	648354	19-04-1995
				FI	A	946194	30-12-1994
				FI	A0	946194	30-12-1994
				MX	A1	9303456	28-02-1994
				NO	A0	945054	27-12-1994
				NO	A	945054	06-02-1995
WO	A1	9621189	11-07-1996	WO	A1	9401820	20-01-1994
				AU	A1	46929/96	24-07-1996
				US	A	5680549	21-10-1997
EP	A1	814403	29-12-1997	US	A	6003063	14-12-1999
				AU	A1	10109/97	17-07-1997
EP	A2	784271	16-07-1997	AU	B2	710044	09-09-1999
EP	A3	784271	21-01-1998	CA	AA	2194520	13-07-1997
				JP	A2	9190411	22-07-1997
				US	A	5948056	07-09-1999